



普通高等教育“十一五”国家级规划教材

C++: 面向对象程序设计 (第2版)

C++: Object-oriented Programming (Second Edition)

李涛 杨剑波 主编



高等教育出版社
HIGHER EDUCATION PRESS

普通高等教育“十一五”国家级规划教材

C++：面向对象程序设计

C++: Mianxiang Duixiang Chengxu Sheji

(第2版)

李 涛 杨剑波 主编

游洪跃 陈良银 李 琳 等编著



高等教育出版社·北京
HIGHER EDUCATION PRESS BEIJING

内容提要

本书全面系统地介绍了面向对象程序设计的基本思想、机制、要素和原理,详细介绍了 C++面向对象程序设计技术;对 MFC 应用程序框架作了专门介绍,力图在介绍基本原理的同时满足动手训练的需要。书后附 Visual C++6.0 编译器使用方法和两套 C++面向对象程序设计模拟试题,供读者学习本课程使用。本书将 C++的基本原理与具体实践相结合,不仅让读者对 C++语言本身有深刻的了解,也为读者迅速掌握流行的 VC++程序设计工具打下了良好的基础。

本书取材新颖、内容丰富、可读性强,所设计的示例程序立足于方便读者观察 C++相关机制的运行过程。

本书可作为高等院校计算机类专业本科生和研究生的教材,亦可供其他从事软件开发工作的读者参考。

图书在版编目(CIP)数据

C++: 面向对象程序设计 / 李涛, 杨剑波主编. --
2 版. --北京: 高等教育出版社, 2013.6

ISBN 978-7-04-037679-1

I. ①C… II. ①李… ②杨… III. ①
C 语言-程序设计-高等学校-教材 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2013) 第 124812 号

策划编辑 倪文慧
插图绘制 尹 莉

责任编辑 倪文慧
责任校对 刘春萍

封面设计 王 洋
责任印制 张泽业

版式设计 马敬茹

出版发行 高等教育出版社
社 址 北京市西城区德外大街4号
邮政编码 100120
印 刷 北京市四季青青青印刷厂
开 本 787mm×1092mm 1/16
印 张 25.25
字 数 570千字
购书热线 010-58581118
咨询电话 400-810-0598

网 址 <http://www.hep.edu.cn>
<http://www.hep.com.cn>
网上订购 <http://www.landracom.com>
<http://www.landracom.com.cn>
版 次 2006年2月第1版
2013年6月第2版
印 次 2013年6月第1次印刷
定 价 35.00元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换
版权所有 侵权必究
物 料 号 37679-00

前 言

本书在第一版的基础上修订完成。本次修订在保持第一版主要内容的基础上,增设专门章节详细介绍标准模板库(STL),并对全书的主要内容进行大幅度修改,以增强内容的连贯性、可读性与实践性,同时简化教学内容,以更符合高校高等教育的教改方向,适应教改洪流,满足加强学生动手能力、创新能力培养力度的整体社会需要。

本书第一版出版后受到读者好评,被列入普通高等教育“十一五”国家级规划教材。但由于方方面面的原因,书中不可避免地存在一些缺陷和错漏。本次改版对这些缺陷和错漏进行了全面修订,使内容描述更清楚、语言更简洁、查错更严格,缺陷进一步减少。感谢同行在使用本书过程中提出的众多中肯建议,新版教材几乎全部采纳了读者和同行的建议和指正。

全书共9章。其中,第1章阐述了C++基础知识,重点介绍了C++和C的主要不同;第2章介绍了面向对象程序设计的基本方法,及C++类的定义、对象的创建等基本内容;第3章着重介绍了C++的继承机制;第4章介绍了C++的多态和重载,并重点介绍了虚函数和运算符重载;第5章介绍了C++的函数模板和类模板机制及其应用;第6章对标准模板库(STL)进行了全面介绍,这是应广大读者的要求新增加的内容;第7章介绍了C++的输入输出流机制;第8章介绍了C++的一些高级主题,如动态内存管理、异常处理和命名空间等;第9章介绍了Visual C++应用程序编程框架,主要介绍MFC编程等。附录包括Visual C++6.0编译器使用方法及两套C++面向对象程序设计模拟试题,供读者学习使用。

本书将提供全面的教学支持,程序源代码及相关教学课件等教学资源将提交中国高校计算机课程网(<http://computer.cncourse.com>)供教师免费下载使用。

本书由李涛(四川大学)和杨剑波(成都理工大学)主编、统稿。书中第1~9章由游洪跃、陈良银、李琳、张靖宇、程艳红、胡大裘编写,田星、吴琼、金军等编写了附录与第9章的部分内容。

特别感谢四川省教学名师唐宁九教授,他在百忙中对本书进行了全面细致的审读,并根据多年从事C++教学的经验对本书的章节安排和内容布局提出了非常中肯的修改意见,使本书的内容更适合教学需求,质量得到进一步提高。

感谢郑万刚和庞廷海两位同学,他们在酷暑难当的夏天对本书的全部内容进行了认真细致的校读,并验证了所有程序。感谢他们的认真工作,他们对事、对人的认真态度给老师们留下了深刻的印象。

感谢柴治、陈黎、金虎、殷锋、王莉、邢建川、王红军、成飏、蔡彪、李宝林、何先

波、倪建成、孙继荣、袁平、卓碧华、欧鹏、周鹏等同行们，他们经常与作者交流教学过程中的心得和体会，并将在使用本书过程中发现的一些优点和不足成文交给作者，为作者编写本书提供了大量的第一手材料。

感谢主管教学的彭舰和洪枚两位副院长、教学科栾新成和何丹两位科长在本书写作过程中所提供的帮助和支持。他们的教学理念、教学要求都深深地影响了本书的成文风格。

由于作者水平有限，新版教材难免仍然存在缺憾与遗漏之处，恳请广大读者不吝赐教，以便再版时修正。

编 者

2013 年 4 月

第一版前言

C++教学已经进行了多年，传统的 C++教学过分注重解释 C++语言本身，而忽视了其在具体环境中的使用。目前高校 C++教学的普遍结果是：学生学完了 C++，但却不会使用目前流行的 C++开发工具编写程序。

基于这一现状，本书力图在介绍 C++面向对象程序设计基本原理及要素的过程中，结合具体实际，详细阐述 C++的主流开发工具 VC++及其程序设计技术，做到理论联系实际，使学生不仅对 C++语言有较深入的理解，同时也能掌握 VC++的基本开发技术，为日后从事软件开发工作打下良好的基础。

另外，有感于部分教材深奥难读的现状，本教材力求通俗易懂，在示例程序的基础上介绍 C++的基本概念和机制，示例程序的设置力求以浅显易懂的方式解释 C++的要素、概念和相关机制。但这并不是说本书没有深度，恰恰相反，本书力求把大部分 C++教程未涉及的内容（比如指针参数传递、动态内存等）以读者容易理解的方式介绍给大家，希望读者学完本书后能够编写出让自己称心如意的程序并激发出设计程序的激情。

全书共分为 10 章。第 1 章阐述了面向对象程序设计的基本原理，重点介绍了 C++语言的历史及其基本架构；第 2 章介绍了 C++类的定义、对象的创建以及对象成员的访问等基本内容；第 3 章着重介绍了 C++的继承机制；第 4 章介绍了多态和重载，重点介绍了虚函数和运算符重载；第 5 章介绍了 C++的模板技术，并在模板技术中对重载、继承和多态机制进行了探讨。第 6 章对 C++中的一些典型问题进行了深入阐述。长期以来，我们感觉这部分内容在教学中会占据较多的课时，但如果在前面的章节中介绍了这些内容，学生就会较难进入面向对象思维模式，当学生具有了面向对象的思维习惯以后再来介绍这些典型问题应该比较合适，本章部分内容有一定难度，但对于一个 C++编程者来说，这些内容都是不可回避的。从第 7 章开始进入 MFC 的学习。第 7 章介绍了 Visual C++编程基础，包括 Windows 程序编程机制、API 编程以及 MFC 编程；第 8 章介绍了对话框、常用消息、菜单和工具条的使用；第 9 章介绍了单文档界面和多文档界面；第 10 章简单介绍了图形设备接口。

本书具有较全面的教学支持，如需本书相关的教案、课件等内容，可通过电子邮件用实名与作者联系。

学生不能很好地适应就业市场的需求，长期以来一直是学生、学校乃至整个社会都非常关注的问题。有感于此，我们对本书的写作思路作了一次开拓性的尝试，希望有利于整个计算机教学的改革。我们不奢望它能成为一本最好的 C++教材，而希望它是一块砖，能够起到

抛砖引玉的作用，引发大家写出更适合读者的教程，为我国计算机人才的培养尽一份自己的力量。

本书第 1、2、3 章由李琳编写，第 4、5、6 章由陈良银编写，第 7、8、9、10 章由游洪跃编写，附录由陈良银与游洪跃共同编写，全书由李涛主编、统稿。

我们要特别感谢唐宁九教授、沈琳副教授，他们为本书的编撰以及精品课程的申报提供了大量的相关资料、素材和宝贵的指导意见。在此特向他们表示最诚挚的谢意。本书还要特别感谢李志蜀教授、谢汶副教授、朱敏副教授和洪枚副教授，感谢他们在百忙中审读本书书稿并提出了许多宝贵的意见。

由于笔者才疏学浅，书中难免有不妥之处，肯请不吝赐教，以便再版时修正。

编 者

2005 年 10 月 18 日

郑重声明

高等教育出版社依法对本书享有专有出版权。任何未经许可的复制、销售行为均违反《中华人民共和国著作权法》，其为人将承担相应的民事责任和行政责任；构成犯罪的，将被依法追究刑事责任。为了维护市场秩序，保护读者的合法权益，避免读者误用盗版书造成不良后果，我社将配合行政执法部门和司法机关对违法犯罪的单位和个人进行严厉打击。社会各界人士如发现上述侵权行为，希望及时举报，本社将奖励举报有功人员。

反盗版举报电话 (010)58581897 58582371 58581879

反盗版举报传真 (010)82086060

反盗版举报邮箱 dd@hep.com.cn

通信地址 北京市西城区德外大街4号 高等教育出版社法务部

邮政编码 100120

目 录

第 1 章 C++基础知识	1
1.1 C++语言简介	1
1.1.1 C++的诞生和发展	1
1.1.2 C++的基本特性	2
1.2 C++对 C 的扩充	2
1.2.1 新的输入输出方式	2
1.2.2 注释、变量声明与作用域限定符	4
1.2.3 强制类型转换与 const 标识符	6
1.2.4 函数原型声明	7
1.2.5 内联函数	7
1.2.6 函数重载	8
1.2.7 带默认参数的函数	9
1.2.8 引用	10
1.2.9 运算符 new 和 delete	12
1.2.10 布尔类型	13
1.3 C++程序的开发过程	13
1.3.1 一个简单的 C++程序	13
1.3.2 C++程序的开发	15
习题 1	15
第 2 章 C++类和对象	16
2.1 面向对象程序设计方法概述	16
2.1.1 面向对象程序设计方法的基本概念	16
2.1.2 面向对象程序设计方法的基本原理	17
2.2 类和对象	19
2.2.1 类的定义	20
2.2.2 成员函数的实现	22
2.2.3 对象的创建	23
2.2.4 类成员的访问	23
2.3 构造函数和析构函数	26

2.3.1	构造函数	26
2.3.2	重载构造函数	30
2.3.3	带默认参数的构造函数	33
2.3.4	拷贝构造函数	35
2.3.5	析构函数	37
2.4	对象数组	40
2.5	对象指针	43
2.5.1	指向对象的指针	43
2.5.2	this 指针	45
2.6	静态成员	49
2.6.1	静态数据成员	49
2.6.2	静态成员函数	52
2.7	友元关系	55
2.7.1	友元函数	56
2.7.2	友元类	59
2.8	共享数据的保护	64
2.8.1	常引用	64
2.8.2	常对象	65
2.8.3	常成员	67
*2.9	应用实例: 员工信息管理系统	71
习题 2		73
第 3 章 继承		74
3.1	类的继承与派生	74
3.1.1	继承与派生的概念	74
3.1.2	派生类的定义	77
3.2	继承方式	78
3.2.1	公有继承	78
3.2.2	私有继承	81
3.2.3	保护继承	84
3.3	派生类的构造函数和析构函数	89
3.3.1	派生类的构造函数	89
3.3.2	析构函数	91
3.4	多继承	93
3.4.1	多继承的声明	93
3.4.2	多继承的构造函数	95

3.4.3 多继承中的二义性问题	96
3.5 虚基类	99
3.5.1 虚基类的作用	99
3.5.2 虚基类的初始化	102
3.6 组合类	104
3.6.1 组合类的概念	104
3.6.2 组合类中的构造函数	105
*3.7 继承应用实例	109
3.7.1 问题描述	109
3.7.2 完整源程序	109
3.7.3 运行结果	116
习题 3	118
第 4 章 多态性	122
4.1 多态性	122
4.2 虚函数	122
4.2.1 虚函数的概念	122
4.2.2 必须使用虚函数的例子	123
4.2.3 虚函数的覆盖	135
4.2.4 虚函数的传递性	139
4.2.5 虚析构造函数	146
4.3 抽象类	148
4.3.1 纯虚函数	148
4.3.2 抽象类	149
4.3.3 使用抽象类的例子	150
4.4 函数重载	154
4.4.1 成员函数重载	155
4.4.2 全局函数重载	157
4.4.3 函数的默认参数	159
4.4.4 二义性问题	160
4.4.5 虚函数与函数重载的关系	160
4.5 运算符重载	161
4.5.1 运算符重载的意义	161
4.5.2 运算符重载的规则	161
4.5.3 成员和非成员运算符函数重载	162
4.5.4 单目和双目运算符函数重载	163

4.5.5	赋值运算符重载和拷贝构造函数	168
4.5.6	其他运算符重载	170
*4.5.7	综合举例	172
习题 4		184
第 5 章	模板	186
5.1	模板的基本概念和作用	186
5.2	函数模板及模板函数	187
5.2.1	函数模板定义及其实例化	187
5.2.2	重载函数模板	189
5.3	类模板及模板类	192
5.3.1	类模板定义及其实例化	192
5.3.2	类模板的派生	198
*5.4	模板实例	199
习题 5		209
*第 6 章	标准模板库	210
6.1	标准模板库概述	210
6.1.1	标准模板库发展简介	210
6.1.2	标准模板库的主要内容	210
6.2	迭代器	211
6.2.1	迭代器的起源	211
6.2.2	迭代器类模板位置	224
6.2.3	迭代器的编程思想	228
6.3	函数对象	228
6.3.1	从函数到函数对象	228
6.3.2	系统函数对象类模板	233
6.4	常用通用容器	237
6.4.1	vector 容器	237
6.4.2	list 容器	240
6.4.3	deque 容器	244
6.4.4	queue 和 stack 容器	246
6.5	STL 常用算法	249
6.5.1	遍历容器算法 for_each()	249
6.5.2	查询算法 find()	253
6.5.3	排序算法 sort()	256
6.6	应用实例：列车调度	259

6.6.1 问题描述	259
6.6.2 分析	259
6.6.3 实现步骤	260
6.6.4 运行程序	263
习题 6	264
第 7 章 C++输入/输出	265
7.1 输入流与输出流基础	265
7.2 标准输出流对象 cout	266
7.3 标准输入流对象 cin	267
7.4 文件操作与文件流	273
7.4.1 文件和文件流基础	273
7.4.2 文件的打开与关闭操作	274
7.4.3 文本文件	275
7.4.4 二进制文件	280
*7.5 应用实例：扩展 C++	284
7.5.1 问题描述	284
7.5.2 分析	285
7.5.3 实现步骤	290
7.5.4 运行程序	298
习题 7	299
第 8 章 C++高级主题	301
8.1 动态内存管理	301
8.1.1 new 运算符和 delete 运算符的作用	301
8.1.2 重载 new 和 delete 运算符	306
8.1.3 常见动态内存错误	308
8.1.4 指针和数组的差异	309
8.1.5 利用指针传递内存	310
8.2 类型转换	313
8.3 异常处理	316
8.4 命名空间	322
习题 8	325
*第 9 章 MFC 开发	326
9.1 MFC 简介	326
9.2 MFC 的主要类	326
9.2.1 MFC 类库中的主要类	326

9.2.2 可视对象类.....	327
9.3 MFC 的消息与消息处理方法	329
9.4 应用程序向导 AppWizard 与类向导 ClassWizard	333
9.4.1 应用程序向导 AppWizard	333
9.4.2 类向导 ClassWizard	339
*9.5 应用实例：人事管理.....	347
9.5.1 问题描述	347
9.5.2 分析	347
9.5.3 实现步骤	348
9.5.4 运行程序	356
习题 9	358
附录	359
附录 A Visual C++ 6.0 编译器使用方法	359
附录 B C++面向对象程序设计模拟试题.....	365
B.1 模拟试题一	365
B.2 模拟试题一参考答案	373
B.3 模拟试题二	377
B.4 模拟试题二参考答案	384
参考文献	387

第 1 章 C++基础知识

1.1 C++语言简介

1.1.1 C++的诞生和发展

C++是从 C 语言发展演变而来的，因此在介绍 C++之前还要先回顾一下 C 语言。C 语言最初是贝尔实验室的 Dennis Ritchie 在 B 语言基础上开发的。它具有许多优点，比如语言简洁灵活，程序执行效率高，同时具有高级语言和汇编语言的优点，等等。由于与其他高级语言相比，C 语言可以直接访问物理地址；与汇编语言相比，它又具有良好的可读性和可移植性，所以 C 语言得到极为广泛的应用。尽管如此，C 语言毕竟是一门面向过程的编程语言，当软件以产业化的规模蓬勃发展的时候，C 语言已经不能满足市场的需求，于是 C++便在 C 语言的基础上应运而生了。

C++是由贝尔实验室的 Bjarne Stroustrup 博士设计和实现的，它兼具 Simula 语言的面向对象程序设计特性和 C 语言的优势。C++最初的版本被称做“带类的 C”，在 1980 年第一次被投入使用；当时它只支持系统程序设计和数据抽象技术。直到 1983 年才逐渐添加了面向对象程序设计的思想方法和语言元素。1985 年，C++第一次投入市场并逐渐取得商业成功。随着越来越多的程序员选择使用 C++来开发产品，正式的 C++标准化工作在 1990 年启动。标准化工作由美国国家标准学会 ANSI 以及后来加入的国际标准化委员会 ISO 负责。1998 年正式发布国际标准 C++ 98，随后在 2003 年更新为 C++ 03，目前最新的 C++标准是 2011 年 9 月 1 日 ISO 发布的 C++ 11。

C++语言的发展历史大致可以概括为以下三个阶段。

第一阶段从 20 世纪 80 年代初期到 1995 年。这一阶段 C++语言基本上传统类型上的面向对象语言，并且凭借接近 C 语言效率的优势，在众多开发语言中占据相当大的份额。

第二阶段从 1995 年到 2000 年。这一阶段由于标准模板库 STL 和后来的 Boost 等程序库的出现，泛型程序设计在 C++中占据了越来越大的比重。当然，同时由于 Java、C#等语言的出现和硬件价格的大规模下降，C++受到了一定的冲击。

第三阶段从 2000 年至今。由于以 Loki、MPL 等程序库为代表的产生式编程和模板元编程的出现，C++出现了发展历史上的又一个高峰。这些新技术的出现以及和原有技术的融合，使 C++成为当今主流程序设计语言中最复杂的一员。

1.1.2 C++的基本特性

C++语言既保留了C语言的有效性、灵活性、便于移植等特点和精华，又添加了对面向对象编程的支持，具有强大的编程功能，可方便地构造出模拟现实问题的实体和操作；编写出的程序具有结构清晰、易于扩充等优良特性，适合于各种应用软件、系统软件的程序设计。用C++编写的程序可读性好，生成的代码质量高，运行效率仅比汇编语言慢10%~20%。

具体来说，C++语言具有以下一些基本特性。

(1) C++是C语言的超集。它既保持了C语言的简洁、高效和接近汇编语言等特点，又克服了C语言的缺点，其编译系统能检查更多的语法错误，因此，C++比C语言更安全。

(2) C++保持了与C语言的兼容。绝大多数C语言程序可以不经修改直接在C++环境中运行，用C语言编写的众多库函数可以用于C++程序中。

(3) C++支持面向对象程序设计。在C++中，所有模拟现实问题的实体和操作都封装在类里，由类可以创建出若干个具有相同属性和行为的对象，并且每个对象都能够接收信息、处理数据和向其他对象发送信息。但是由于C++与C的兼容性，它同时还支持面向过程的程序设计，因此，C++不能算是一门纯粹的面向对象语言。

(4) C++程序在可重用性、可扩展性、可维护性和可靠性等方面都较C语言得到了提高，使其更适合开发大中型的系统软件 and 应用程序。

1.2 C++对C的扩充

除了增加面向对象的编程方案外，C++对C语言的许多方面进行了扩充与改进。下面进行简要的介绍。

1.2.1 新的输入输出方式

在C++中，程序的输入输出仍可使用C语言中的库函数，如printf()、scanf()等，但同时也引入了新的输入输出方式，即使用系统预定义的I/O流类对象。

C++预定义了I/O流类对象供用户进行标准I/O操作，主要包含cin和cout。其中cin是istream类的对象，也称为标准输入流对象，它是从标准输入设备（键盘）获取数据，程序中的变量通过提取符“>>”从流中提取数据；cout是ostream类的对象，也称为标准输出流对象，它是向标准输出设备（显示器）输出数据，程序中的变量通过插入符“<<”向流中插入数据。使用cin和cout时，必须包含头文件iostream。

1. cout

与cout相配合，C++提供了一个新的运算符“<<”用于输出。运算符“<<”称为插入运算符，意思是将输出数据插入到输出流中。

说明:

(1) 可以在一行输出语句中连续使用“<<”，用以连续向屏幕输出信息。

(2) 对于一般的指针，“<<”运算符输出其指针值；但对于字符型指针，“<<”运算符输出其所指向的字符串。

(3) 可以直接输出字符串常量，它将原样显示在屏幕上。

(4) 可以使用格式控制符 endl 来控制输出换行，相当于C语言中的转义符'\n'。

例 1.1 C++输出示例。

```
//文件路径名:s1_1\smain1_1.cpp
#include <iostream>
using namespace std;

int main()
{
    int i=10;
    char c='a';
    float f=20.325;
    int *pi=&i;
    char *pa="Hellow World!";
    cout<<i<<'\\t'<<c<<'\\t'<<f<<endl;
    cout<<pi<<'\\t'<<pa<<endl;
    cout<<"这是一个串常量! ";
    return 0;                //返回值 0, 返回操作系统
}
```

程序的运行结果为:

```
10   a   20.325
0012FF38   Hellow World!
这是一个字符串常量!
```

2. cin

与 cin 相配合，C++提供了一个新的运算符“>>”用于输入，运算符“>>”称为提取运算符，意思是从输入字节流中取出一个数据。可以在一行输入语句中连续使用“>>”，用以连续输入多个数据，“>>”运算符从左向右依次运算。

例 1.2 C++输入示例。

```
//文件路径名:s1_2\smain1_2.cpp
#include <iostream>
using namespace std;

int main()
{
```