



高等学校计算机教育“十二五”规划教材 ■■■

C++程序设计教程

董小园 主编

任建强 王希更 殷秀莉 刘强 副主编

++ CHENGXU
SHEJI JIAOCHENG

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

高等学校计算机教育“十二五”规划教材

C++程序设计教程

董小园 主 编

任建强 王希更 殷秀莉 刘 强 副主编

赵姝菊 冯淑杰 王肖林 崔 文 参 编

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书从 C++ 语言的基础开始讲解, 围绕其面向对象的主旨和内涵由浅入深层层展开, 将 C++ 语法知识及应用程序的开发贯穿其中, 使学生打下 C++ 面向对象程序设计的根基。全书共 13 章, 各章配以丰富完整的实例程序, 教学生编写面向对象程序及可重复使用的类。书中的程序除了配有详细的代码说明外, 更注意建立程序设计的观念, 让学习者通过轻松的学习流程, 学会 C++ 的语法, 借助完整的程序解析, 学会范例的应用程序的设计, 进一步了解 C++ 面向对象程序设计的方法、经验, 完全掌握 C++ 程序设计。

本书行文流畅, 实例丰富, 描述细致严谨, 既适合作为高等院校相关专业的 C++ 程序设计课程的教材, 也适合作为初学 C++ 语言或希望掌握面向对象编程思想的读者自学使用。

图书在版编目 (CIP) 数据

C++ 程序设计教程 / 董小园主编. — 北京: 中国铁道出版社, 2012. 8

高等学校计算机教育“十二五”规划教材

ISBN 978-7-113-15046-4

I. ①C… II. ①董… III. ①C 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 (2012) 第 154928 号

书 名: C++ 程序设计教程

作 者: 董小园 主编

策 划: 秦绪好

读者热线: 400-668-0820

责任编辑: 赵 鑫

编辑助理: 赵 迎

封面设计: 刘 颖

责任印制: 李 佳

出版发行: 中国铁道出版社 (100054, 北京市西城区右安门西街 8 号)

网 址: <http://www.51eds.com>

印 刷: 北京海淀五色花印刷厂

版 次: 2012 年 8 月第 1 版 2012 年 8 月第 1 次印刷

开 本: 787 mm×1 092 mm 1/16 印张: 14.75 字数: 346 千

印 数: 1~3 000 册

书 号: ISBN 978-7-113-15046-4

定 价: 29.00 元

版权所有 侵权必究

凡购买铁道版图书, 如有印制质量问题, 请与本社教材图书营销部联系调换。电话: (010) 63550836

打击盗版举报电话: (010) 63549504

高等学校计算机教育“十二五”规划教材

主 任：陈 明

副主任：叶曲伟 严晓舟

委 员：（按姓氏笔画排序）

王希更	王智广	王锁柱	刘贵龙	李 环
李海生	李素朵	张晓明	陈志泊	陈晓云
赵裕民	郝 莹	侯耀军	姚 琳	秦绪好
袁东光	袁 薇	徐孝凯	徐 慧	郭成芳
曹永存	解 凯	管建和		

序言

PREFACE

随着计算机科学与技术的飞速发展，现代计算机系统的功能越来越强大，应用也越来越广泛，尤其是快速发展的计算机网络。它不仅是连接计算机的桥梁，而且已成为扩展计算能力、提供公共计算服务的平台，计算机科学对人类社会的发展做出了卓越的贡献。

计算机科学与技术的广泛应用是推动计算机学科发展的原动力。计算机科学是一门应用科学。因此，计算机学科的优秀创新人才不仅应具有坚实的理论基础，还应具有将理论与实践相结合来解决实际问题的能力。培养计算机学科的创新人才是社会的需要，是国民经济发展的需要。

计算机学科的发展呈现出学科内涵宽泛化、分支相对独立化、社会需求多样化、专业规模巨大化和计算教育大众化等特点。一方面，使得计算机企业成为朝阳企业，软件公司、网络公司等 IT 企业的数量和规模越来越大，另一方面，对计算机人才的需求规格也发生了巨大变化。在大学中，单一计算机精英型教育培养的人才已不能满足实际需要，社会需要大量的具有职业特征的计算机应用型人才。

计算机应用型教育的培养目标可以利用知识、能力和素质三个基本要素来描述。知识是基础、载体和表现形式，从根本上影响着能力和素质。学习知识的目的是为了获得能力和不断地提升能力。能力和素质的培养必须通过知识传授来实现，能力和素质也必须通过知识来表现。能力是核心，是人才特征的最突出的表现。计算机学科人才应具备计算思维能力、算法设计与分析能力、程序设计能力和系统能力(系统的认知、设计、开发和应用)。计算机应用型教育对人才培养的能力要求主要包括应用能力和通用能力。应用能力主要是指用所学知识解决专业实际问题的能力；通用能力表现为跨职业能力，并不是具体的专业能力和职业技能，而是对不同职业的适应能力。计算机应用型教育培养的人才所应具备的三种通用能力是学习能力、工作能力、创新能力。基本素质是指具有良好的公民道德和职业道德，具有合格的政治思想素养，遵守计算机法规和法律，具有人文、科学素养和良好的职业素质等。计算机应用型人才素质主要是指工作的基本素质，且要求在从业中必须具备责任意识，能够对自己职责范围内的工作认真负责地完成。

计算机应用型教育课程类型分为通用课程、专业基础课程、专业核心课程、专业选修课程、应用课程、实验课程、实践课程。课程是载体，是实现培养目标的重要手段。教育理念的实现必须借助于课程来完成。本系列规划教材的特点是重点突出、理论够用、注重应用，内容先进、实用。

本系列教材的不足之处，敬请各位专家、老师和广大同学指正。



2012年3月

本书的编写主旨是为 C++ 语言的初学者提供一套有效的学习资料。书中充分考虑到读者在学习过程中可能出现的问题和感受，提供大量典型实例及经验总结，带领读者通过不断的程序编写掌握 C++ 语言的核心思想及应用。希望本书能够在读者学习、工作上给予读者最大程度的帮助！

全书共 13 章，主要内容如下：

- 第 1 章 C++ 语言概述，带领读者近距离认识 C++，对 C++ 的含义、特点、工作方式等进行介绍。

- 第 2 章 变量与数据类型。

- 第 3 章 运算符。

- 第 4 章 选择与循环，对 C++ 程序的结构、基本语法、结构化语句进行详尽的描述。

- 第 5 章 函数，介绍 C++ 中非常重要的函数相关知识，以及如何灵活应用函数。

- 第 6 章 数组，介绍数组的定义和使用。

- 第 7 章 结构体与枚举，介绍两种新的自定义类型，并带领读者进行练习掌握其使用。

- 第 8 章 指针，介绍地址的概念，这是 C++ 很有特点的、十分重要的知识点之一。

将面向对象思想的学习带入更高层次：继承和多态。

- 第 9 章 重载，介绍 C++ 中函数的灵活应用，并体现了面向对象的多态特性。

- 第 10 章 类与对象、第 11 章 类的继承，集中讲授 C++ 面向对象的内容，并对封装、继承、多态这三大特性进行说明和举例。

- 第 12 章 成员函数的其他特性，补充了友元函数、虚函数等知识，使 C++ 程序的编写更加丰富而精彩。

- 第 13 章 文件与流，介绍输入/输出、文件读写的相关知识。

本书主要特点如下：

1. 注重实践、例题丰富

本书配有大量程序例题，并对实例做了详细的说明。

每章各知识点均有丰富的典型例程。全书的所有实例都尽量做到内容易懂、特点突出。编写、运行并理解书中所有的实例将引导读者完成全书的学习。

2. 资料完善、辅助学习

本书每章都配有上机实验，书中还详细介绍了开发环境 Visual Studio 2005 的安装和使用说明，带领读者使用主流开发工具，在实操中掌握编程理论。同时还提供全书配套的 PPT 讲义。

3. 描述精准、生动易懂

编者在写书时尽量考虑到读者在阅读及学习过程中可能出现的问题和感受，以

通俗易懂的语言对内容进行叙述，并使用比喻、举例、图、表等多种方式对较抽象的知识点进行描述。书中还总结了大量操作步骤、注意事项、内容要点，帮助读者以最直观高效的方式掌握各章节内容精髓。

本书既适合作为高等院校相关专业的 C++程序设计课程的教材，也适合作为初学 C++语言或希望掌握面向对象编程思想的读者自学使用。

由于编者水平有限，加之时间仓促，书中如有不妥之处，请读者不吝指正。

编 者

2012年4月

目 录

CONTENTS

第 1 章 C++语言概述	1
1.1 C++与面向对象	1
1.2 Visual Studio 2005 集成开发环境	2
1.3 基本的程序结构	3
1.3.1 函数	3
1.3.2 输出与输入	4
1.3.3 预处理程序	4
1.3.4 程序的注释	5
1.4 编写第一个程序	5
小结	10
上机实验	11
第 2 章 变量与数据类型	12
2.1 浅谈数据类型	12
2.2 整数类型	13
2.2.1 整数类型分类	13
2.2.2 整数类型的修饰符	14
2.3 字符类型	15
2.4 浮点类型	16
2.4.1 浮点类型分类	16
2.4.2 浮点类型与整型的比较	17
2.5 布尔类型	17
2.6 变量	18
2.6.1 变量的声明	18
2.6.2 变量的初始化	19
2.7 常量	20
2.7.1 使用 const 声明常量	20
2.7.2 使用#define 声明常量	20
2.8 变量与常量的生命周期	21
2.8.1 局部变量	21
2.8.2 全局变量	22
2.8.3 自动变量	22
2.8.4 静态变量	23
2.9 转换数据类型	23
2.9.1 自动转换	24
2.9.2 强制转换	25
小结	26
上机实验	27

第3章 运算符	28
3.1 赋值运算符	28
3.2 算术运算符	29
3.2.1 基本算术运算符	29
3.2.2 更简洁的语句方式	31
3.2.3 自增自减运算符	31
3.3 关系运算符	33
3.4 逻辑运算符	34
3.5 其他运算符	36
3.5.1 条件运算符	36
3.5.2 逗号运算符	37
3.5.3 求字节数运算符	37
3.6 运算符的优先级	37
小结	38
上机实验	38
第4章 选择和循环	39
4.1 if 选择结构语句	39
4.1.1 单分支选择结构	39
4.1.2 双分支选择结构	41
4.1.3 多分支选择结构	42
4.2 switch 语句	45
4.3 while 循环	47
4.4 do...while 循环	49
4.5 for 循环	51
4.6 break 语句和 continue 语句	52
4.6.1 break 语句的使用	52
4.6.2 continue 语句的使用	54
小结	55
上机实验	56
第5章 函数	57
5.1 理解函数	57
5.1.1 函数的声明	57
5.1.2 函数的定义	58
5.1.3 调用函数	60
5.2 参数与返回值	65
5.2.1 声明带参数与返回值的函数	65
5.2.2 定义带参数与返回值的函数	66
5.2.3 调用函数时传入参数	67
5.2.4 函数返回值	68
5.3 传址参数	73
5.4 默认参数值的函数	75

5.5	内联函数	76
5.6	变量的种类	77
5.6.1	自动变量与静态变量	77
5.6.2	局部变量与全局变量	79
	小结	80
	上机实验	81
第 6 章	数组	82
6.1	理解数组	82
6.2	一维数组	83
6.2.1	一维数组的声明	83
6.2.2	一维数组的初始化	83
6.2.3	一维数组元素的使用	84
6.3	多维数组	87
6.3.1	二维数组的声明	87
6.3.2	二维数组元素访问	88
6.4	将数组作为函数参数	89
6.5	字符串	92
6.5.1	使用字符数组表示字符串	92
6.5.2	字符串的使用	93
6.5.3	string 类	95
6.6	数组的应用	97
6.6.1	排序	98
6.6.2	逆转字符串	100
	小结	102
	上机实验	102
第 7 章	结构体与枚举	103
7.1	结构体的定义	103
7.1.1	定义结构体的语法	103
7.1.2	声明结构类型的变量	104
7.1.3	定义结构体常见的问题	105
7.2	结构体成员的访问	106
7.3	传递结构体变量给函数	109
7.4	函数返回结构变量	111
7.5	结构体数组	113
7.6	枚举类型	114
7.6.1	枚举的定义	114
7.6.2	枚举的使用	115
	小结	115
	上机实验	116
第 8 章	指针	117
8.1	理解指针	117

8.1.1	内存地址	117
8.1.2	地址运算符“&”	118
8.2	指针变量	119
8.2.1	指针变量的声明	119
8.2.2	指针变量的初始化	120
8.2.3	指针变量的使用	120
8.2.4	void 指针	122
8.3	指针与函数	123
8.3.1	传递指针作为参数	123
8.3.2	返回指针	127
8.4	指针与结构体	128
8.4.1	结构体类型指针的声明	128
8.4.2	结构体类型指针访问成员变量	128
8.5	指针与数组	129
8.5.1	指针与数组声明	129
8.5.2	指针与数组元素	129
8.5.3	指针与数组参数	131
	小结	135
	上机实验	135
第 9 章	重载	137
9.1	函数的重载	137
9.1.1	函数重载的概念	137
9.1.2	以参数个数不同实现函数重载	138
9.1.3	以参数类型不同实现函数重载	140
9.2	运算符的重载	142
9.2.1	运算符重载的语法	143
9.2.2	运算符重载的参数	144
9.2.3	运算符重载的返回值	144
9.2.4	重载一元运算符	145
9.2.5	重载二元运算符	148
9.2.6	重载 cin 与 cout	152
	小结	156
	上机实验	156
第 10 章	类与对象	157
10.1	类与对象的概念	157
10.1.1	类与结构体	157
10.1.2	类与对象	158
10.2	类的成员函数	160
10.3	构造函数	164
10.3.1	构造函数的概念	164
10.3.2	带有参数的构造函数	166

10.3.3	构造函数的重载	168
10.4	对象的复制	170
10.5	析构函数	171
10.5.1	析构函数的概念	172
10.5.2	析构函数的使用	172
	小结	175
	上机实验	175
第 11 章	类的继承	177
11.1	继承的概念	177
11.2	基类与派生类	178
11.2.1	继承的语法	178
11.2.2	重载成员	180
11.2.3	派生类的构造函数	183
11.2.4	继承与类的转换	186
11.3	访问控制	186
11.4	多重继承	187
11.4.1	多重继承的语法	187
11.4.2	多重继承的问题	188
	小结	188
	上机实验	189
第 12 章	成员函数的其他特性	190
12.1	virtual 函数	190
12.1.1	改变基类函数	191
12.1.2	使用 virtual 函数	193
12.2	抽象类	195
12.2.1	抽象类的概念	195
12.2.2	抽象类的使用	196
12.3	friend 函数	198
12.3.1	friend 函数的概念	198
12.3.2	friend 函数的使用	199
12.4	static 静态成员	201
12.4.1	static 静态成员变量	201
12.4.2	static 成员函数	203
	小结	206
	上机实验	206
第 13 章	文件与流	207
13.1	流类	207
13.1.1	ios 类	207
13.1.2	ostream 类	208
13.1.3	istream 类	210
13.2	流错误	210

13.2.1	状态位	210
13.2.2	状态函数	211
13.3	文件 I/O	212
13.3.1	写入文件数据	213
13.3.2	读取文件数据	214
13.3.3	读取包含空格的字符串	214
13.3.4	检测文件结尾	215
13.3.5	关闭文件	215
13.3.6	文件的读写与数据格式	216
13.4	对象 I/O	218
	小结	220
	上机实验	221
	参考文献	222

第1章

C++语言概述

本章将介绍 C++ 的一般特性。C++ 不仅具备高级语言易学、易用的特点，还具备低级语言高效率的特性，同时，C++ 的面向对象特点，让原本烦琐的程序趋于简单化。

本章还介绍 C++ 的编辑环境和一些必备的知识，从了解 C++ 的开发环境、熟悉 C++ 程序的开发流程到第一个 C++ 程序的编写，来一探 C++ 的奥秘。

学习目标

- 理解什么是面向对象
- 了解面向对象的优点
- 熟悉 Visual Studio 2005 的程序编辑环境
- 掌握 C++ 程序的基本结构
- 熟练使用 Visual Studio 2005 编写并运行第一个 C++ 程序

1.1 C++与面向对象

C++ 语言源于 C 语言，几乎任何 C 语言中正确的语句都可以在 C++ 中使用，但它们又有本质上的不同：C++ 是一种成熟的面向对象程序设计语言。

面向对象程序设计简称 OOP (Object Oriented Programming)，其主旨是“基于对象的编程”。对象是对现实世界实体的模拟，因此可以更容易地去分析需求，人们可以把万事万物都看做各种不同的对象。面向对象程序设计将事物的共同性质抽象出来，使用数据和方法描述对象的状态和行为。

这就像现实生活中开公司，如果采用传统的结构化分析与设计方法，那么开公司的这个人就要考虑每天先做什么、再做什么，所有事务都得亲自过问，还要跨行业去处理事务，如财务、人事、行政等，结果必定是很累而且效率很低。如果采用面向对象的思想，就先分析好公司正常运营都需要哪些部门、涉及哪些资源，每岗位的要求和职能是什么，然后按照需求聘用人员、准备资源，每个人按各自职能办事，相互合作。这样，不但效率高，还能及时进行局部调整，公司一定经营得井井有条。这里的各部门人员和资源就是对象，把对象都定义好了，需要时让对象们各自发挥作用就可以了。

以上分析不难看出，面向对象程序设计使人们的编程与实际的世界更加接近，所有的对象被赋予属性和方法，编程就更加富有人性化。

面向对象程序设计达到了软件工程的3个主要目标：重用性、灵活性和扩展性。

面向对象的编程方法强调对象的“抽象”“封装”“继承”“多态”，它们是面向对象编程的核心，将在后续章节一一讲解。

1.2 Visual Studio 2005 集成开发环境

C++程序的编辑环境有很多，编写完成的程序可以通过环境提供的编译器进行转换，形成计算机能够识别、执行的程序。本书采用 Microsoft 公司的 Visual Studio 2005（以下简称 VS 2005）进行 C++程序的编写，VS 2005 是一套可视化的 C++程序编辑软件，集成程序编写、测试、运行所需的多种工具，容易使用。目前市面上还有 Visual Studio 2008 和 Visual Studio 2010 等版本，但从占用空间、运行速度和稳定性等方面考虑，2005 版本依然是开发者使用最多的版本。

VS 2005 集成开发环境操作方便，功能实用，程序启动速度快，提供人性化的功能。VS 2005 利用通俗易懂的可视化分割方式，将环境分成数个局部，包含一个简单的文档窗口、一个隐含许多编辑功能的菜单栏、一个功能齐全的工具栏等，让程序设计人员可以轻易地在这个环境下编写程序，以及对程序进行链接、编辑及编译的工作。

VS 2005 的主窗口界面如图 1-1 所示。



图 1-1 VS 2005 主窗口

标题栏位于窗口顶端，显示工程名称。在标题栏最左端的是窗口控制菜单框，标题栏最右端的是最大化按钮、最小化按钮和关闭按钮。

菜单栏中的主菜单及其子菜单提供了 VS 2005 集成开发环境的所有功能。菜单栏除了包括常用的文件、编辑、视图和帮助等菜单外，还有 VS 2005 解决方案相关的项目、生成及调试等菜单。

工具栏提供了与菜单栏中常用的菜单命令相对应的命令按钮，从而达到快捷操作的目的。

解决方案窗口主要用于显示解决方案、解决方案的项目及这些项目中的项。可以在其中对项

目文件进行添加、打开、复制、删除等一系列操作。只需要右击相应的选项，在弹出的快捷菜单中进行选择即可。

文档窗口是 VS 2005 中最重要的功能窗口之一，用于程序代码编辑。VS 2005 提供了符合国家标准、具有严格验证的提示功能代码编辑环境，当程序员在文档窗口中编写代码时，编辑器会自动识别程序语法而显示不同的文字颜色，让程序人员可以清楚地识别程序指令，使编辑工作变得更轻松。

输出窗口显示程序调试或运行过程中的状态和结果。

使用 VS 2005 进行 C++ 程序开发的具体步骤将在 1.4 节详细介绍。

1.3 基本的程序结构

编译器能够根据程序结构正确地识别程序源代码，并编译成可以由计算机执行的文件，因此编写程序时，必须遵守程序语言的结构与规则，才能让设计的程序正常地被编译、执行。这里先来认识一个 C++ 程序基本结构中的各主要部分。

1.3.1 函数

函数是 C++ 程序中最基本的程序架构，可以说 C++ 程序是由许许多多的“函数”组成的，即使是最简单的程序也会包含一个函数——main 函数。每个函数代表着一段抽象出来的语句，能够实现某些功能，其意义在于功能的独立和反复使用。函数必须有自己的名称，让编译器可以加以识别，并在需要的时候反复调用。

关于函数的具体内容将在相应章节具体描述，这里先简要介绍一下 main 函数。

每个 C++ 的主要程序代码都必须放在 main 函数里，如果程序中没有这个函数，即使有再多的代码，程序还是无法执行，main 函数就像 C++ 程序的“心脏”一样，决定着程序能否执行。

main 函数不能更改名称，编译器在编译程序时，会先从 main 函数开始编译，如果没有这个函数，就无法完成编译的工作。main 函数的结构如下：

```
void main()
{
    //函数的起始位置
    //程序语句;
    ...
}
//函数的结束位置
```

函数的语句区必须包括在左大括号“{”和右大括号“}”中，左大括号表示函数的起始位置，右大括号表示函数的结束位置。函数有多种写法，包括有“返回类型”的函数，有“参数”的函数等，用来传递调用函数时所需的信息。

“返回类型”表示函数返回值的数据类型，写在函数名称的前面，例如：

```
int main()
```

上述写法表示调用 main 函数后，main 函数必须返回一个整型类型（int）的数值。又如：

```
void main()
```

返回类型为 void，表示该函数不返回任何值。

对于有返回类型的函数，在函数结束前必须加入 return 语句，用来返回该数据类型的值，return 0 表示结束执行的程序并返回操作系统，格式如下：

```
int main()
```

```
{
```

```
...
```