

C++

程序设计

主 编 王 超

副主编 陈文斌 周 军 曹庆旭



- 本书注重内容的实用性，重点培养应用能力
- 系统介绍C++面向对象程序设计的基本概念、基本语法和编程方法
- 提供大量实例、习题和实验，增强读者动手编程能力
- 本书吸收了国内外教材的优点，凝集了编者多年的教学和软件项目开发的经验，易学易用
- 教学光盘的内容丰富实用，可以大大提高学习效率 and 教学效果

地质出版社

21世纪高校计算机应用系列规划教材

C++程序设计

主 编 王 超

副主编 陈文斌 周 军 曹庆旭

地 质 出 版 社

· 北 京 ·

内 容 提 要

欢迎加入学习C++程序设计的队伍。C++语言是现今非常流行的程序设计语言，深受广大程序员的青睐。本教材从简单的样例入手，边学边练，重点讲述解决问题的思路；精心设计每一章节的样例，由浅入深、讲解透彻；每章备有相关练习题，让读者能及时巩固。

本教材前8章介绍C++语言的基础知识，这部分内容包括：C++语言基础、逻辑运算、控制语句、数组、函数、编译预处理、指针、结构体。后8章介绍C++语言面向对象程序设计，这部分内容包括：C++语言的新特性、类与对象、静态成员、友元、运算符重载、继承、多态性和虚函数、流、模板。

此外，本教材所有的例程都在Visual C++ 6.0下调试通过，读者在学习本教材的过程中，还可以学习如何在Visual C++ 6.0中编写控制台程序。

本书配有多媒体教学光盘，将理论学习中难于理解的内容用生动形象的多媒体形式展现在读者面前，使学习变得更加轻松。

本教材适合作为高等院校相关专业的教材使用，也适合于程序设计的初学者，可供成人高考和在职人员培训使用。

图书在版编目(CIP)数据

C++程序设计/王超主编. —北京：地质出版社，
2006.8

ISBN 7-116-04713-1

I. C... II. 王... III. C语言—程序设计
IV. TP312

中国版本图书馆CIP数据核字(2006)第036206号

C++ CHENG XU SHE JI

责任编辑：邱红 刘丹

责任校对：郑淑艳

出版发行：地质出版社

社址邮编：北京海淀区学院路31号，100083

电 话：(010) 82324508 (邮购部)

网 址：<http://www.gph.com.cn>

电子信箱：zbs@gph.com.cn

传 真：(010) 82310759

印 刷：北京盛兰兄弟印刷装订有限公司

开 本：787mm×1092mm 1/16

印 张：24.25

字 数：601千字

版 次：2006年8月北京第1版 2006年8月第1次印刷

定 价：32.00元

ISBN 7-116-04713-1/T 125

(凡购买地质出版社的图书，如有缺页、倒页、脱页者，本社出版处负责调换)

为普及计算机技术做贡献

原清华大学校长 张孝文 书赠
国家教委副主任

丛书序言

在我国高等教育逐步实现大众化后，高等院校的教育模式也逐渐面向国民经济发展的第一线，为行业、企业培养各级各类高级应用型专门人才。为大力推广计算机应用技术，更好地满足我国高等院校从精英教育向大众化教育的转变，符合社会对高等院校应用型人才培养的要求，北京洪恩教育科技有限公司组织成立了“21世纪高校计算机应用系列规划教材编委会”，在明确了高等院校应用型人才培养模式、培养目标、教学内容和课程体系的前提下，组织编写了本套“21世纪高校计算机应用系列规划教材”。

众所周知，教材建设作为保证和提高教学质量的重要支柱及基础，作为体现教学内容和教学方法的知识载体，在当前培养应用型人才中的作用是显而易见的。然而，目前市场上的电脑图书虽然种类繁多，但与教学相宜的教材很少。因此，本套教材是编委会经过对近千所高等院校和上百家知名企业的调研后，组织全国近百所院校的骨干教师和数十位不同领域的工程师在广泛交流和研讨的基础上编写的。教材的编者都是来自从事计算机教学的一线教师和就职于各知名企业的工程师，以及长期从事知名多媒体电脑教学软件——《开天辟地》、《万事无忧》、《畅通无阻》和《巧夺天工》等教学研究和开发的电脑专家，具有非常丰富的教学和实践经验。

以下是本系列教材的主要特点：

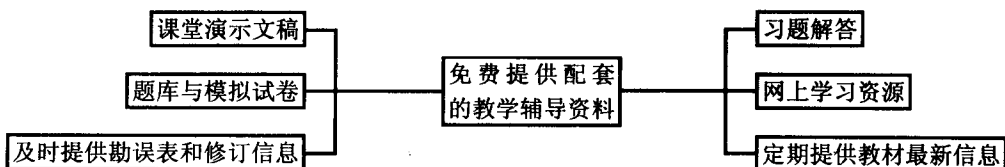
(1) 突出应用技术，全面针对实际应用。在选材上，根据实际应用的需要，坚决舍弃现在用不上、将来也用不到的内容。在保证学科体系完整的基础上不过度强调理论的深度和难度，注重应用型人才的专业技能和工程实用技术的培养。

(2) 教材采用“任务驱动”的编写方式，采取“提出问题——介绍解决问题的方法——归纳总结，培养寻找答案的思维方法”的模式。以实际问题引导出相关原理和概念，在讲述实例的过程中将知识点融入，通过分析归纳，介绍解决工程实际问题的思想和方法，然后进行概括总结，使教材内容层次清晰，脉络分明，可读性和操作性强。同时，引入案例教学和启发式教学方法，便于激发学习兴趣。

(3) 在教材内容编排上，力求由浅入深，循序渐进，举一反三，突出重点，运用口语化的语言，通俗易懂，讲求效率，内容经过多次提炼和升华，突出学习规律和学习技巧，是思维化的直接体现。

(4) 在形式上，充分发挥了洪恩教育在多媒体领域的独特优势，每本教材均有配套的多媒体教学，开创了国内把多媒体教学引入高校的教学领域的先例。直观讲解和演示操作的全过程，使学习效率更高；难点问题用动画演示，使教学更轻松。另外，我们还同步提供相关的配套教辅，如课堂内外的学习辅导、实验指导、综合培训、课程设计指导等。

(5) 提供立体化服务。



为方便教学，我们将为选用本系列教材的老师免费提供PowerPoint电子教案、Flash课件、习题解答、题库和模拟试卷等，并及时提供教材的前沿信息，使教材向多元化、多

媒体化发展,最大限度地满足广大教师进行多媒体教学的需要。此外,还免费提供相关教材中所有程序的源代码或教学素材,以提高教学效率。

选用本书作教材的任课老师可以拨打电话010-58858208或通过洪恩在线的教材素材专区(<http://pcbook.hongen.com>)下载或发邮件到pcbook@goldhuman.com信箱免费索取PowerPoint电子教案、Flash课件、习题解答、题库或模拟试卷等相关资料。

总之,本套教材凝聚了众多长期在教学、科研一线工作的老师和数十位软件工程师的经验和智慧。我们感谢该套教材的各位作者为教材出版所做的贡献,也感谢姜波、李洪旺、刘玉兴、帅立松、徐润、杨文海等为丛书编辑和其他工作所付出的努力。

脚踏实地、精益求精;科教兴国、行胜于言。洪恩软件永远与您在一起。我们期待广大读者对本套规划教材提出宝贵意见,以便进一步修订,使该套规划教材不断完善。

编委会

2006年8月

21世纪高校计算机应用系列规划教材

编委会名单

主任:池宇峰

副主任:李宏明 卢志勇 姜天鹏

委员:(以下排名按姓氏字母的先后顺序为序)

陈常晖 董锡臣 耿杰 何黎明 胡荣 江玉珍

隽青龙 李虹 李军 李晓松 李亦明 林立宇

刘丽君 刘文毓 刘泽云 卢继阳 潘全春 彭铁光

邵晓光 谭晓芳 万宏辉 王科 王改香 王巧莲

吴艳华 谢枫 姚小兵 查玉祥 张燕丽 赵志芳

前 言

C++语言是现今非常流行的程序设计语言，深受广大程序员的青睐。本教材从简单的样例入手，边学边练，重点讲述解决问题的思路；精心设计每一章节的样例，由浅入深、讲解透彻；每章备有相关练习题，让读者能及时巩固。

本教材前8章介绍C++语言的基础知识，这部分内容包括：C++语言基础、逻辑运算、控制语句、数组、函数、编译预处理、指针、结构体。后8章介绍C++语言面向对象程序设计，这部分内容包括：C++语言的新特性、类与对象、静态成员、友元、运算符重载、继承、多态性和虚函数、流、模板。

此外，本教材所有的例程都在Visual C++ 6.0下调试通过，读者在学习本教材的过程中，还可以学习如何在Visual C++ 6.0中编写控制台程序。

本书配有多媒体教学光盘，将理论学习中难于理解的内容用生动形象的多媒体形式展现在读者面前，使学习变得更加轻松。

本教材适合作为高等院校相关专业的教材使用，也适合于程序设计的初学者，可供成人高考和在职人员培训使用。

本书由王超主编，陈文斌、周军、曹庆旭副主编，其中第1、2、3、7、8章由王超编写，第4~5章由周军编写，第6、9、10章由陈文斌编写，第11~13章由曹庆旭编写，参加编写和审校等工作的还有林立宇、李虹、宋红、赵方和李晓松等。

编 者

2006年8月

目 录

第1章 C++语言概述及Visual C++ 6.0控制台程序

1.1 编程语言的发展	1
1.1.1 编程和编程语言	1
1.1.2 编程语言的发展史	2
1.2 面向对象语言简介	2
1.2.1 面向对象的概念	2
1.2.2 面向对象语言的特点	4
1.3 面向过程与面向对象的区别	4
1.3.1 面向过程方法面临的问题	4
1.3.2 面向对象方法的特点	5
1.3.3 算法	6
1.4 C与C++语言的关系	7
1.4.1 C++对C语言的改进	7
1.4.2 C++语言对C语言扩充和增强的几点具体体现	7
1.5 C++语言结构上的特点	9
1.6 C++程序的实现	11
1.6.1 C++程序的编辑、编译和运行	12
1.6.2 编译出现的问题	12
1.6.3 Visual C++ 6.0编译系统的用法简介	13
1.7 程序是如何执行的	17
1.8 注意C中字母的大小有别	17
1.9 C++语言的知识结构图	18

第2章 基本数据类型、运算符和表达式

2.1 基本数据类型	21
2.1.1 常量和变量	21
2.1.2 C++语言的基本数据类型及其表示	24
2.2 运算符和表达式	31
2.2.1 给变量赋初值	31
2.2.2 运算符和表达式	32
2.2.3 数据类型的转换	37
2.3 C++的输入和输出	39
2.3.1 输出流对象cout使用介绍	39

2.3.2 输入流对象cin使用介绍	41
练习	42
第3章 C++语言基础部分	
3.1 程序的三种基本结构	45
3.1.1 顺序结构	45
3.1.2 选择结构	46
3.1.3 循环结构	47
3.1.4 程序的流程图	48
3.2 选择控制语句	48
3.2.1 关系运算符和关系表达式	49
3.2.2 逻辑运算符和逻辑表达式	50
3.2.3 条件运算符和条件表达式	51
3.2.4 条件选择语句if	53
3.2.5 分支结构switch	58
3.3 循环控制语句	62
3.3.1 for循环语句	62
3.3.2 while循环语句	65
3.3.3 do while循环语句	67
3.3.4 多重循环语句	68
3.3.5 break语句和continue语句	69
练习	72
第4章 数组	
4.1 一维数组	75
4.1.1 一维数组的定义	75
4.1.2 一维数组元素的赋值和初始化	76
4.1.3 一维数组的程序举例	79
4.2 二维数组	84
4.2.1 二维数组的定义	84
4.2.2 二维数组元素的赋值和初始化	85
4.2.3 二维数组程序举例	87
4.3 字符数组	89
4.3.1 字符数组的定义	89
4.3.2 字符数组的初始化	89
4.4 字符串和字符串处理	90

4.4.1 字符串的定义	90
4.4.2 字符串的初始化	91
4.4.3 字符串的输入输出	91
4.4.4 字符串处理函数	93
练习	103
第5章 函数和存储类	
5.1 函数的定义与调用	108
5.1.1 什么是函数	108
5.1.2 函数的定义	109
5.1.3 函数的声明	115
5.1.4 函数的调用	115
5.2 参数传递	121
5.3 递归函数	122
5.4 内联函数	124
5.5 函数原型	128
5.6 重载函数	128
5.7 缺省参数	130
5.8 变量的作用域与生存期	131
5.8.1 变量的作用域	131
5.8.2 变量的生存期和存储类型	133
练习	137
第6章 编译预处理	
6.1 宏定义	141
6.1.1 不带参数的宏定义	141
6.1.2 带参数的宏定义	143
6.2 文件包含	144
6.3 多文件程序	145
练习	146
第7章 指针和引用	
7.1 指针	148
7.1.1 什么是指针	148
7.1.2 指针的定义	150
7.1.3 指针的运算	155

7.2 指针与数组	158
7.2.1 指针与数组的关系	158
7.2.2 应用举例	161
7.2.3 指针数组	167
7.3 指针、数组与函数	170
7.3.1 指针作函数的参数	170
7.3.2 数组与函数	173
7.3.3 用变量的地址作为函数的实参	175
7.3.4 返回指针的函数	176
7.4 利用指针动态分配内存	177
7.5 引用	181
7.5.1 独立引用	182
7.5.2 引用作为参数传递	183
7.5.3 引用作为返回类型	186
7.5.4 引用与指针的区别	187
练习	188
第8章 结构体和枚举类型	
8.1 结构体类型	193
8.1.1 什么是结构体	193
8.1.2 结构体的定义	194
8.1.3 定义结构体变量	195
8.1.4 结构体变量的初始化	198
8.1.5 结构体数组	201
8.1.6 结构体指针	203
8.1.7 结构体与函数	207
8.2 枚举类型	212
8.2.1 枚举类型的定义	212
8.2.2 枚举变量的声明	213
8.2.3 枚举类型变量的赋值和使用	213
8.2.4 枚举类型的应用举例	214
练习	217
第9章 类与对象	
9.1 类和对象	220
9.1.1 新的数据类型“类”的引入	220

9.1.2 类和对象	221
9.1.3 类的定义和实现	222
9.1.4 类的使用	223
9.1.5 类的定义及实现示例程序演示	224
9.1.6 提示：在Visual C++ 6.0的程序中添加一个类	225
9.2 存取访问控制	229
9.2.1 关键字public	229
9.2.2 关键字private	230
9.2.3 关键字protected	233
9.3 类与结构体	233
9.4 应用举例	234
练习	238
第10章 深入了解“类”	
10.1 new和delete	239
10.1.1 new	240
10.1.2 delete	241
10.2 构造函数和析构函数	242
10.2.1 构造函数	242
10.2.2 析构函数	251
10.3 再谈new、delete和malloc、free	253
10.4 this指针	255
10.5 应用举例	258
练习	265
第11章 静态成员和友元	
11.1 静态成员	267
11.1.1 静态数据成员	267
11.1.2 静态成员函数	271
11.2 友元	273
11.2.1 友元函数	274
11.2.2 友元类	276
11.3 应用举例	277
练习	280

第12章 运算符重载

12.1 明确目标：为什么要进行运算符重载	281
12.2 以成员函数实现运算符重载	282
12.2.1 重载一元运算符：递增运算符“++”	283
12.2.2 二元运算符重载：重载加运算符“+”	291
12.2.3 重载赋值运算符“=”	294
12.3 用友元函数重载运算符	297
12.3.1 用友元函数重载加法运算符“+”	297
12.3.2 重载输出运算符“<<”	300
12.4 应用举例	302
练 习	310

第13章 继 承

13.1 什么是继承	311
13.2 实现继承	313
13.2.1 单一继承	313
13.2.2 多重继承	318
13.2.3 多级继承	321
13.3 构造函数、析构函数与继承	322
13.3.1 何时执行构造函数和析构函数	322
13.3.2 将参数传送给基类的构造函数	324
13.4 继承与友元及静态成员的关系	329
13.4.1 友元与继承	329
13.4.2 静态成员与继承	331
13.5 应用举例	332
练 习	337

第14章 虚函数与多态性

14.1 虚函数	339
14.1.1 为什么要引入虚函数	339
14.1.2 虚函数与函数重载的区别	345
14.1.3 虚函数与继承的关系	346
14.1.4 虚函数的层次性	346
14.1.5 析构函数与虚函数	348
14.2 纯虚函数	349
练 习	351

第15章 流和文件	
15.1 格式化输出	352
15.1.1 I/O (输入/输出) 控制符介绍.....	352
15.1.2 格式化输出举例	353
15.2 文件流	356
15.2.1 文件的打开和关闭	356
15.2.2 文件读写操作举例	357
练习	361
第16章 模板	
16.1 函数模板	362
16.2 类模板	365
练习	370
附录A ASC II 表	371
附录B 常用库函数	372

第1章 C++语言概述及Visual C++ 6.0控制台程序



教学目标

初步认识C++语言，了解编程语言的发展史。掌握C++语言的编程特点。熟悉C++语言的语法规则以及使用Visual C++ 6.0来编译和运行C++代码。



教学重点和难点

- ◆ C++语言的框架
- ◆ main()函数、注释语句
- ◆ 用Visual C++ 6.0编写、编译、执行一个C程序

C++是一种面向对象的程序设计语言，它是在C语言的基础上发展起来的。虽然它不是最早的面向对象的程序设计语言，但它是目前使用较为广泛的面向对象的程序设计语言。

本章主要讲述C++语言的编程特点、C++语言与C语言的关系、C++语言的语法规则以及C++语言的实现。在讲述C++语言之前，介绍一个与C++语言有关的重要概念——面向对象，包括什么是面向对象、面向对象语言的特点、面向对象和面向过程的区别等，通过对面向对象概念的学习和理解，将会进一步了解C++语言的特点及其用途。

1.1 编程语言的发展

1.1.1 编程和编程语言

早期人们认为编程工作包含有认识事物和描述事物两项内容，认为编程应先对要解决的问题产生一个正确的认识，然后再用一种语言将它正确地描述出来。这样就把软件开发与编程看成一回事了。从软件工程的角度来看，软件开发和编程是不同的。软件开发包含两项主要活动：一是对人们要解决的问题的认识；二是对这种认识的描述。“认识”是指对要解决的问题进行周密地分析和全面理解，并找出解决的方法；“描述”是指选用一种语言来描述对要解决问题的认识。可见，编程只是在认识基础上进行的描述，编程时所选用的语言称为编程语言。因此，我们认为编程只是软件开发中的一项内容，而不是全部内容。

开发人员对于要解决的问题的认识又称为对问题域的认识。问题域是指要解决问题的集合，或者指要解决的问题所涉及的业务范围。人们对于问题域的认识往往用自然语言来描述，而计算机所能识别的却是某种编程语言。于是在自然语言和编程语言之间存在一个过渡，或称为“鸿沟”。这个“鸿沟”形成的原因很简单，就是因为人们描述客观事物所用自然语言机器不能识别，而机器能够识别（直接或间接）的编程语言又不符合人们习惯的思维方式。于是就形成了二者之间的鸿沟。鸿沟的存在耗费软件开发人员的许多精力，同时也是许多错误的发源地。

1.1.2 编程语言的发展史

编程语言是从低级到高级发展的，具体过程如下：

1. 机器语言

这是最原始的编程语言，这种语言是计算机可以直接识别的语言。它使用0和1两种代码，编写出的程序难以理解和记忆，因为它远离人们习惯的思维方式。

2. 汇编语言

这是一种使用助记符号来代替代码0和1，是一种低级语言。它比机器语言稍有提高，较为符合人们的形象思维，它是低层次的抽象。计算机不能直接识别汇编语言，需要编译后才可识别。这种语言虽然效率较高，但是由于难以记忆，使用较少。

3. 高级语言

这是一种采用命令或语句的语言，屏蔽了机器细节问题。它提高了语言的抽象层次，比汇编语言更加接近于人们的思维方式。这种语言人们容易理解和记忆，但它还是与自然语言有较大差别。20世纪70年代结构化程序设计语言的出现，给编程带来了方便，使得自然语言与编程语言的差距进一步缩小。

4. 面向对象语言

面向对象语言是比面向过程语言更高级的一种高级语言。

面向对象语言的出现改变了编程者的思维方式，使设计程序的出发点由着眼于问题域中的过程转向着眼于问题域中的对象及其相互关系。这种转变更加符合人们对客观事物的认识，因此，面向对象语言更接近于自然语言，面向对象语言是人们对于客观事物更高层次的抽象。

从编程语言发展的历史来看，编程语言由低级向高级发展，使得自然语言与编程语言之间的联系越来越紧密，这就意味着软件开发人员的精力耗费越来越少，软件产品的质量越来越高。

1.2 面向对象语言简介

本节主要讲述计算机出现后编程语言的发展简史。从这段历史中不难看出面向对象语言的出现是人们对客观事物认识不断发展的需要，因此，面向对象语言的出现是必然的。本节首先介绍面向对象这一概念。

1.2.1 面向对象的概念

什么是面向对象？简单地说，它和面向过程一样都是软件开发的一种方法。但是它与面向过程不同，面向对象是一种运用对象、类、继承、封装、聚合、消息传递、多态性等概念来构造系统软件的软件开发方法。这里提出一些新的概念，这些新的概念描述了面向对象的新特点。

下面将从解释这些概念的过程中给出面向对象的特点，进而对面向对象这种方法有所了解 and 认识。

1. 对象是软件系统的基本构成单位

分析问题的出发点是对象，对象是待解决问题中的客观事物的抽象表示。

2. 对象的属性和服务结合为一个独立的实体

对象的属性是表示客观事物的静态特性，一般用数据来表示；对象的服务是描述客观事物的动态特性，即事物的行为，一般用函数或称方法来表达。对象的属性与服务结合为一个独立的整体，称为封装体，对外屏蔽其内部的细节。

3. 类是对某些对象的描述

类是具有相同属性和服务的若干对象的集合。类为该类的所有对象提供一种统一的抽象描述。一个类中包含有属性和服务两部分。实际上，类是一种类型，这种类型是自定义的，而对象是某个类的实例。

4. 派生类继承基类中的属性和服务

在不同的层次上运用抽象的原则，可以获得基类和它的派生类。而派生类继承了基类中的属性和服务，这将简化系统的构造过程和文档，实现了共享。例如，根据世界上存在的各种各样的汽车，可以将它们的共性抽象为一个汽车类，再将各种各样的轿车的共性抽象为一个轿车类。显然轿车类应该是汽车基类的派生类，因为轿车具有汽车的共性，另外还具有它自身的特性。我们说轿车类继承了汽车类中的属性和服务。

5. 繁杂对象可由简单对象构成

复杂的事物可以分解为若干简单的事物。同样，一个复杂的对象可以分解为简单对象的集合，这称为聚合或聚集。例如，一架飞机，这是一个较为复杂的对象，但可以将它看成由机翼、机身、发动机和尾翼等部件组成，这些部件可作为飞机的子对象。

6. 对象与对象之间使用消息进行通信

信息是对象发出的服务请求，信息的发送是一个要求提供服务的对象，而信息的接收者是一个能够提供服务的对象；通过消息传送实现了对象之间的通信。

7. 多态性是面向对象程序设计的重要支柱

多态性是指向不同对象发送同一消息，根据不同的对象完成不同的行为。多态性是通过继承的机制构造对象类的结果。同函数和运算符的重载以及虚函数实现了类的多态性。

以上是对面向对象这一概念的理解。下面将通过一个例子形象地说明封装的概念。例如，大街上的早点小吃店，在一间小屋或小亭子里，四周被围住，通过一个窗口对外卖货。屋或亭内有油饼、油条、包子、豆浆等食品，另外，还具有制作这些食品的服务，如炸油饼、油条等，还有收钱找零等服务。其实我们可以将早点店看作是一个封装体，买早点的人们只关心自己感兴趣的早点和价格，至于这些早点怎么制作，人们并不关心。这个被封装的“实体”对外服务只通过一个“窗口”，当买早点的人向它发出“买什么早点”、“买多少”、“付钱”等消息后，它就将早点递给你。把具有这类特点的服务称作早点服务类，而某家早点店看成为早点服务类的一个对象。

综上所述，面向对象这种方法具有三大特性：封装性、继承性和多态性。特别是封装性和继承性是不可缺少的。