

高等学校计算机专业教材

GAODENG XUOXIAO JISUANJI ZHUANYE JIAOCAI

面向对象的程序设计语言 ——C++ (第二版)

◎ 陈志泊 主编 王春玲 孟伟 编著

GAODENG XUOXIAO JISUANJI ZHUANYE JIAOCAI
GAODENG XUOXIAO JISUANJI ZHUANYE JIAOCAI
GAODENG XUOXIAO JISUANJI ZHUANYE JIAOCAI
GAODENG XUOXIAO JISUANJI ZHUANYE JIAOCAI

人民邮电出版社
POSTS & TELECOM PRESS

高等学校计算机专业教材

面向对象的程序设计

语言——C++

(第二版)

陈志泊 主编

王春玲 孟 伟 编著

人民邮电出版社

北 京

图书在版编目 (CIP) 数据

面向对象的程序设计语言: C++ / 陈志泊主编. —2 版. —北京: 人民邮电出版社, 2007.8
高等学校计算机专业教材

ISBN 978-7-115-16110-9

I. 面... II. 陈... III. C 语言—程序设计—高等学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2007) 第 054833 号

高等学校计算机专业教材

面向对象的程序设计语言——C++ (第二版)

-
- ◆ 主 编 陈志泊
编 著 王春玲 孟 伟
责任编辑 滑 玉
 - ◆ 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>
三河市海波印务有限公司印刷
新华书店总店北京发行所经销
 - ◆ 开本: 787×1092 1/16
印张: 20
字数: 480 千字 2007 年 8 月第 2 版
印数: 29 001—32 000 册 2007 年 8 月河北第 1 次印刷

ISBN 978-7-115-16110-9/TP

定价: 28.00 元

读者服务热线: (010)67170985 印装质量热线: (010)67129223

内 容 提 要

本书是《面向对象的程序设计语言——C++》一书的第二版。全书内容主要分成两大部分。

第一部分是 C++ 语言基础部分，包括第 1 章～第 5 章，主要讲述 Visual C++ 6.0 集成开发环境的使用、Windows 控制台应用程序的开发方法和步骤、数据类型、运算符、表达式、3 种基本程序结构、数组、函数及其重载、引用、指针、结构体、共用体、编译预处理等。通过这一部分的学习，初学者即使没有 C 语言的基础或其他语言的基础，也可以直接使用本书。

第二部分是面向对象的程序设计部分，包括第 6 章～第 12 章，详细讲述面向对象程序设计的有关基本概念、类与对象的定义和使用、构造函数与析构函数、继承与派生、虚函数与多态性、友元函数与友元类、静态成员、模板、文件操作、异常处理等。

全书内容相互衔接，前后呼应，循序渐进。在每一部分内容的讲解中，概念清晰、条理性强、深入浅出，并给出大量的实例进行解释说明。同时，每章后面都附有大量的各种类型的习题。习题答案可到人民邮电出版社网站 (www.ptpress.com.cn) 下载。

为方便读者学习和教师讲课，本书还提供了 PowerPoint 电子讲稿，读者可到人民邮电出版社的网站下载。书中的所有源程序都已在 Visual C++6.0 环境中调试通过。

本书作为大学计算机及相关专业的本、专科程序设计语言课程的教材，也比较适合广大计算机爱好者自学和参考。

编者的话 (第二版)

本书自第一版出版以来,受到了广大读者和计算机爱好者的热烈欢迎。但是,随着计算机技术的飞速发展,新技术、新知识层出不穷,一本教材的内容不可能一成不变,也不可能包罗万象。同时,在第一版的使用过程中,很多读者也给我们提出了很多有益的建议和意见,在这种情况下,我们经过认真分析、讨论,将第一版中的三部分内容进行了合理的取舍,本书主要保留了第一版中的前两部分内容,并进行了大量的修改、补充和完善,尽可能将 C++ 语言、面向对象的编程思想、方法和技术讲透、讲全;而第一版中的第三部分内容即 Visual C++ 部分计划另写其他教材,并与本书形成系列教材。

本书是《面向对象的程序设计语言——C++》一书的第二版。在编写过程中,我们深入调查了目前许多高校讲授 C++ 语言课程的详细情况,同时参阅和借鉴了国内外许多优秀教材的内容和优点,并进一步吸取了第一版教材的优点和教学过程中的体会和经验,与第一版相比,本书体现了以下几方面的特点。

(1) 更加符合初学者学习 C++ 语言的认识规律,进一步提高了概念讲解条理性、逻辑性,内容讲解循序渐进,深入浅出,易于读者学习和掌握。

(2) 进一步扩充了 C++ 语言的基础部分,进一步完善了本书的内容体系,初学者即使没有 C 语言的基础或其他语言的基础,也可以直接使用本书。

(3) 进一步完善和扩充了 C++ 语言的面向对象的程序设计内容,尤其是对类和对象、继承与派生、虚函数与多态性等内容进行了更完善、更全面的讲解。另外,还增加了 C++ 异常处理、String 类和 STL 库等内容。

(4) 进一步丰富了例题,并将所学内容和相关知识点组织到相应例题中,使读者能很容易地进一步加深对知识点学习的理解和掌握。

(5) 进一步丰富了习题的类型和数量,除了简答题、编程题、读程序写结果题之外,还增加了选择题、判断题、填空题、改错题等,从各种不同的侧面进一步帮助读者对所学知识练习和掌握。

(6) 全书所有例题、习题中的程序代码都已在 Visual C++6.0 开发环境中运行通过,保证了代码的正确性。

(7) 增加了 Visual C++ 开发环境的使用和操作、开发程序的方法和步骤,使读者很容易学会利用 Visual C++ 开发环境编写 C++ 应用程序,真正做到学以致用。

本书正是基于这种思想而编写的,全书共分成了两部分内容,第一部分是 C++ 语言基础部分,第二部分是面向对象的程序设计部分。全书内容相互衔接,成为一个逻辑整体。每章除提供大量的实例进行讲解外,还附有一定数量的习题,便于读者练习,习题答案可到人民邮电出版社网站 (www.ptpress.com.cn) 下载。

本书内容全面,深入浅出,概念清晰,条理清楚,不仅适用于教学,也适合读者自学。如果作为教材,建议学时为 70 学时,其中课堂教学 50 学时。由于课程学时的限制,实验学

时各学校可适当调整，一般为 20 学时，另外，除实验学时外，最好安排学生自由上机的时间，加强学生的实际动手能力。与本书配套的 PowerPoint 课件可到人民邮电出版社网站 (www.ptpress.com.cn) 下载。

本书由陈志泊担任主编，全书由王春玲、孟伟编写，由陈志泊修改定稿。其中，第 1 章～第 5 章、第 12 章主要由孟伟编写，第 6 章～第 11 章主要由王春玲编写。另外，张慧萍、黄金燕、冯振兴、王鹏等参加了文字编校、PowerPoint 讲稿制作工作，在此表示衷心的感谢！

由于时间仓促，加之作者水平有限，不当之处在所难免，敬请广大读者批评指正。

编者
2007 年 4 月

编者的话 (第一版)

随着计算机技术的飞速发展及其应用领域的扩大,快速而扎实地掌握和精通一门程序设计语言成为计算机专业人员不可回避的课题。多年来的实践和经验告诉我们,C语言以其代码效率高、灵活性强、运算符丰富、结构化程序设计方法以及与操作系统的结合紧密等特点,受到了广大计算机专业人士及计算机爱好者的欢迎。

但是,随着 Windows 操作系统功能的逐步完善以及面向对象的程序设计方法的提出,程序设计与软件开发的思路、方法、复杂程度都发生了巨大的变化,使得利用 C 语言设计 Windows 的应用程序变得力不从心。而 C++ 语言是在 C 语言的基础上发展起来的,不仅继承了 C 语言的原有特点,而且引入了面向对象的程序设计方法,从而成为计算机及相关专业程序设计语言课程的首选语言。另外,Microsoft 公司推出的 Visual C++6.0 集成开发环境提供了建立控制台应用程序的方法,为我们学习和掌握 C++ 语言提供了非常好的开发环境,同时,它又很好地利用了 C++ 语言,并提供了大量的 MFC (Microsoft Foundation Class) 基础类库、应用程序向导和“可视化”的资源编辑器,这为程序员快速而高效地开发出 Windows 应用程序提供了方便。因此,一方面,Visual C++6.0 集成开发环境使得在学习 C++ 语言和开发 Windows 下的应用程序方面都能兼顾,另一方面,也使我们在开发 Windows 下的应用程序时不再感到繁琐和困难。

然而,目前市面上能全面介绍 C++ 语言及其应用、并能将它们汇编到一本书中的实用教材并不多见。另外,一本教材不能只介绍程序设计语言本身,还必须让读者从中学会程序设计的思想、方法及其在处理实际问题中的用途。本书正是基于这种思想而编写的,全书共分为三部分内容,第一部分是 C++ 语言基础部分,第二部分是面向对象的程序设计部分,第三部分是面向对象的程序设计应用部分。前两部分重点介绍了 C++ 语言的概念、基本语法及面向对象的程序设计的方法,第三部分重点介绍了利用 C++ 语言、面向对象的程序设计方法、Visual C++ 6.0 集成开发环境及 MFC 类设计 Windows 应用程序的方法,这部分内容是对所学 C++ 语言的运用和综合。另外,全书内容相互衔接,成为一个逻辑整体。除第 1 章外,每章都附有一定数量的习题,便于读者学习。为方便读者学习和教师讲课,本书还提供了用 Power Point 制作的讲稿。需要者请到人民邮电出版社的网站中下载(选择“教材出版中心”→“教材附件”→“课件”即可)。

本书内容全面,深入浅出,不仅适合于教学,同时也适合于用 Visual C++ 开发应用程序的用户参考。如果作为教材,建议第一部分和第二部分应全讲,而对于第三部分,各院校可根据学时情况适当选讲,如果学时紧张,第 8 章可选讲。建议主讲学时为 50 学时,学时分配为:第一部分 15 学时,第二部分 20 学时,第三部分 15 学时。由于课程学时的限制,实验学时各学校可适当调整,一般为 20 学时左右;另外,除实验学时外,最好还要安排学生自由上机的时间,以加强学生的实际动手能力。

本书由陈志泊统稿和主编，全书由陈志泊、王春玲编写。另外，张慧萍、黄金燕等参加了文字编辑工作，马振龙参加了书稿校对方面的工作，在此一并表示衷心的感谢！

由于时间仓促，加之作者水平有限，书中不当之处在所难免，敬请读者批评指正。

编者

2002年1月

目 录

第 1 章 C++语言概述	1
1.1 从面向过程的程序设计到面向对象的程序设计	1
1.1.1 传统的结构化程序设计方法	1
1.1.2 面向对象的程序设计方法	3
1.1.3 面向对象的程序设计方法与结构化程序设计方法的比较	7
1.2 C++语言的产生	7
1.3 C++语言的特点	8
1.4 C++程序开发过程	8
1.4.1 Visual C++ 6.0 集成开发环境	8
1.4.2 Visual C++ 6.0 控制台应用程序开发实例	13
1.5 C++程序的结构与基本组成	16
1.5.1 一个简单的 C++程序	16
1.5.2 使用名空间	17
1.5.3 C++输入输出简介	19
1.6 小结	19
习题	19
第 2 章 基本数据类型、运算符与表达式	21
2.1 数据类型	21
2.1.1 基本数据类型	21
2.1.2 类型修饰符	22
2.2 常量和变量	23
2.2.1 常量	23
2.2.2 变量	26
2.3 运算符和表达式	30
2.3.1 算术运算符和算术表达式	30
2.3.2 赋值运算符和赋值表达式	32
2.3.3 sizeof 运算符	33
2.3.4 关系运算符和关系表达式	34
2.3.5 逻辑运算符和逻辑表达式	34
2.3.6 条件运算符	36
2.3.7 位运算符	36
2.3.8 逗号运算符	37
2.3.9 运算符的优先级与结合性	37
2.4 数据的输入与输出	38
2.4.1 数据的输出	38

2.4.2 数据的输入	42
2.5 综合实例	43
2.6 小结	44
习题	44
第3章 C++的控制语句	47
3.1 C++语句概述	47
3.2 C++程序的3种基本结构	48
3.3 if语句	49
3.3.1 单分支if语句	50
3.3.2 双分支if语句	51
3.3.3 多分支if语句	52
3.3.4 if语句的嵌套	52
3.4 switch语句	54
3.5 循环语句	55
3.5.1 while循环语句	55
3.5.2 do-while循环语句	56
3.5.3 for循环语句	57
3.5.4 循环嵌套	59
3.5.5 3种循环的比较	59
3.6 限定转向语句	60
3.6.1 break语句	60
3.6.2 continue语句	60
3.7 综合实例	61
3.8 小结	62
习题	62
第4章 函数	66
4.1 函数的定义	66
4.1.1 函数的定义	66
4.1.2 函数的参数	67
4.1.3 函数的返回值	68
4.2 函数的调用	68
4.2.1 函数的调用形式	68
4.2.2 函数的嵌套调用	70
4.2.3 函数的递归调用	71
4.3 内联函数	73
4.3.1 内联函数的定义方法	73
4.3.2 内联函数与普通函数的区别和联系	73
4.3.3 对内联函数的限制	73
4.4 带默认形参值的函数	74

4.4.1 默认参数的函数	74
4.4.2 使用默认参数的函数的注意事项	74
4.5 函数重载	75
4.5.1 函数重载定义	75
4.5.2 匹配重载函数的顺序	75
4.5.3 定义重载函数时的注意事项	76
4.6 变量的作用域与生存期	76
4.6.1 局部变量	77
4.6.2 静态局部变量	77
4.6.3 全局变量	78
4.6.4 静态全局变量	79
4.7 函数的作用域	80
4.7.1 只能被本文件调用的函数	80
4.7.2 可被其他文件调用的函数	80
4.8 编译预处理	81
4.8.1 宏定义命令	81
4.8.2 文件包含命令	82
4.8.3 条件编译命令	82
4.9 综合实例	84
4.10 小结	87
习题	87
第5章 构造数据类型	91
5.1 数组	91
5.1.1 一维数组的定义与引用	91
5.1.2 二维数组的定义与引用	94
5.1.3 字符数组与字符串	96
5.2 指针	102
5.2.1 指针变量的定义与使用	102
5.2.2 const 指针	104
5.2.3 用指针作为函数的参数	105
5.2.4 数组的指针	106
5.2.5 数组名作函数参数	108
5.2.6 返回指针值的函数	113
5.2.7 函数的指针	114
5.2.8 指针与字符串	115
5.2.9 多重指针	117
5.2.10 动态内存分配	117
5.3 引用	119
5.3.1 引用的概念、声明和使用	119

5.3.2	引用与函数	121
5.3.3	const 引用	124
5.3.4	引用总结	125
5.4	结构体、共用体和枚举	125
5.4.1	结构体	125
5.4.2	共用体	131
5.4.3	枚举类型	133
5.5	类型定义	134
5.6	综合实例	134
5.7	小结	136
	习题	137
第6章	类和对象	141
6.1	类与对象的定义	141
6.1.1	类的定义	141
6.1.2	成员函数的定义	143
6.1.3	对象及指向对象的指针	145
6.1.4	访问对象的成员	146
6.1.5	指向类成员的指针	147
6.1.6	this 指针	150
6.1.7	对象赋值语句	150
6.1.8	对象的作用域与生存期	151
6.2	构造函数与析构函数	152
6.2.1	构造函数	152
6.2.2	构造函数的重载	154
6.2.3	默认构造函数与缺省参数的构造函数	155
6.2.4	析构函数	156
6.2.5	拷贝构造函数	157
6.2.6	浅拷贝与深拷贝	161
6.3	对象成员及其初始化	163
6.3.1	对象成员	163
6.3.2	对象成员的初始化	164
6.4	类作用域	167
6.5	堆对象和对象数组	169
6.5.1	堆对象	169
6.5.2	对象数组	169
6.6	静态成员	171
6.6.1	静态数据成员	171
6.6.2	静态成员函数	173
6.7	友元函数和友元类	177

6.7.1 友元函数	177
6.7.2 友元类	179
6.8 常对象与常成员	181
6.8.1 常对象	181
6.8.2 常成员	181
6.9 综合实例	184
6.10 小结	187
习题	188
第7章 继承和派生	196
7.1 继承和派生的概念	196
7.2 单继承	197
7.2.1 单继承的定义方式	197
7.2.2 派生类的成员构成	199
7.2.3 继承方式对基类成员的访问属性控制	199
7.2.4 派生类的构造函数	206
7.2.5 有子对象的派生类的构造函数	210
7.2.6 派生类的析构函数	211
7.3 多重继承	212
7.3.1 多重继承的定义方式	212
7.3.2 多重继承的二义性	214
7.3.3 虚基类及其派生类的构造函数	216
7.4 虚函数与多态性	220
7.4.1 多态性	220
7.4.2 赋值兼容规则	221
7.4.3 用基类指针指向公有派生类对象	222
7.4.4 虚函数	224
7.4.5 纯虚函数与抽象类	226
7.5 综合实例	228
7.6 小结	234
习题	235
第8章 运算符重载	243
8.1 运算符重载的概念	243
8.2 重载为类的成员函数	243
8.3 重载为类的友元函数	244
8.4 运算符重载的限制	245
8.5 典型运算符重载	246
8.5.1 赋值运算符的重载	246
8.5.2 单目运算符的重载	247
8.6 综合实例	249

8.7 小结	251
习题	251
第9章 模板	253
9.1 类模板	254
9.1.1 类模板的定义	254
9.1.2 类模板的使用	254
9.2 函数模板	256
9.2.1 函数模板的定义	256
9.2.2 函数模板的使用	257
9.3 综合实例	258
9.4 小结	260
习题	261
第10章 文件	263
10.1 通过 FILE 结构进行文件操作	263
10.1.1 打开文件	264
10.1.2 关闭文件	265
10.1.3 文件的读写	266
10.2 通过文件流进行文件操作	269
10.2.1 流类库概述	269
10.2.2 打开文件	270
10.2.3 关闭文件	270
10.2.4 文件的读写	271
10.3 综合实例	273
10.4 小结	275
习题	275
第11章 string 类与 STL 简介	278
11.1 string 类	278
11.1.1 string 类的构造函数	278
11.1.2 string 类的成员函数	279
11.1.3 string 类的运算符	280
11.2 标准模板库	281
11.2.1 范型化程序设计	281
11.2.2 标准模板库	282
11.2.3 容器	282
11.2.4 迭代器	284
11.2.5 算法	287
11.3 综合实例	288
11.4 小结	288
习题	289

第 12 章 异常处理	290
12.1 异常的概念	290
12.2 C++异常处理机制	290
12.2.1 异常处理的语法	291
12.2.2 使用多条 catch 语句	293
12.2.3 捕获所有类型的异常	294
12.3 限制异常与重抛异常	295
12.3.1 限制异常	295
12.3.2 重抛异常	296
12.4 异常处理中对象的构造和析构	296
12.5 综合实例	298
12.6 小结	299
习题	299
附录	301
附录 1 ASCII 字符表	301
附录 2 C++部分库函数	302
参考文献	304

第 1 章 C++ 语言概述

本章内容是面向对象程序设计、C++语言程序开发的基础，在后续各章学习之前，应该很好地掌握。

本章主要内容如下。

- (1) 结构化程序设计方法与面向对象程序设计方法的各自特点、区别和相互联系。
- (2) 面向对象的程序设计方法中的有关概念，如类、对象、封装、继承、消息和多态性等。
- (3) 利用 VC++ 6.0 生成简单应用程序的步骤。
- (4) C++程序的基本组成和框架结构，注意它与 C 程序间的区别和联系。
- (5) 名空间机制以及 C++的输入输出过程。

1.1 从面向过程的程序设计到面向对象的程序设计

1.1.1 传统的结构化程序设计方法

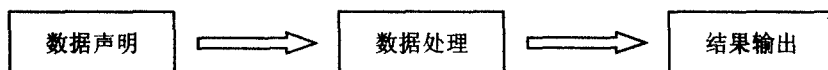
1. 使用传统的结构化程序设计 (Structured Programming, SP) 方法设计程序

使用 SP 方法设计程序时，一般按以下步骤进行。

步骤 1: 根据实际问题的要求对其进行分析，找出解决问题的方法和步骤 (即设计算法)。

步骤 2: 为了在计算机上具体实现出算法，需要为这种算法选择或构造适当的数据结构，通过对数据的操作过程体现算法的思想。也就是说，程序是在数据的某种特定表示方式和结构的基础上对抽象算法的具体描述。

步骤 3: 利用 SP 方法设计程序的结构如下。



其中，在数据处理过程中，采用的是自顶向下、分而治之的方法，将整个程序按功能划分为几个可独立编程的子过程模块，每一子模块完成指定的子任务并且提供一个清晰、严格的调用界面，主过程通过调用各子过程来完成全部处理工作。这样可将一个大型的、复杂的问题分解为若干个小的、易于管理和维护的模块，从而有效地降低了程序设计的复杂性。

【例 1-1】最简单的 C++ 程序。

```
#include <iostream>
```



```
using namespace std;
int main()
{   cout<<"hello world.";           //输出 hello world
    return 0;
}
```

程序的运行结果为

hello world.

下面通过例 1-1 了解一下 C++程序的组成。

(1) 每一个 C++程序都是由若干个函数构成的，其中必须有且仅有一个名称为 main 的函数，它是 C++程序的入口点，程序每次执行，都先从 main 函数开始。

(2) main 表示函数名，前面的 int 表示 main 函数应该返回一个整型的函数值。其后面的函数体由一对花括号括起来，可以包含多条语句，每条语句以分号结束，例 1-1 中包含两条语句。cout 语句是 C++程序中用于输出的语句，主要由 cout 和 << 实现，将 << 右侧双引号中的字符串输出。return 语句是向操作系统返回一个零值。

(3) 程序第一行中的“#include <iostream>”不是 C++的语句，而是 C++的一个预处理，它以“#”开头，行末尾没有分号，称为“包含命令”，作用是将文件 iostream 的内容包含到该命令所在的程序文件中，代替该命令行。文件 iostream 被称为“头文件”，包含了程序实现输入输出操作所需要的信息。在程序进行编译时，先对所有的预处理命令进行处理，用头文件的具体内容代替#include 命令行，然后再对程序进行整体编译。

(4) 程序第二行中的语句“using namespace std;”表示“使用名空间 std”。由于 C++标准库中的类和函数是在名空间 std 中声明的，因此程序中如果用到 C++标准库，就需要用“using namespace std;”进行声明。

C++的初学者可以先记住程序的结构和相关的写法，不必深究，后续章节还会对一些内容进行更详细的分析，逐步加深读者对于 C++程序的理解。

【例 1-2】一个简单的面向过程的 C++程序。

```
#include <iostream>
using namespace std;           //名空间声明
void print();                  //函数声明
void main()
{
    int i;
    char s[80];
    print( );                  //调用函数 print
    cout<<"What's your name?\n"; //用 C++特有的方式输出数据
    cin>>s;                    //用 C++特有的方式输入数据
    cout<<"How old are you?\n";
    cin>>i;
    //验证结果
    cout<<s<<" is "<<i<<" years old.";
}
void print( )
{   cout<<("Personal Information: \n");
}
```

程序的执行结果为