



普通高等教育“十一五”规划教材

C++ 程序设计教程

主编 田秀霞 徐建平 彭源



附赠

- ① 书中涉及的全部程序
- ② 精心制作的精美、实用的教学课件
- ③ 一组精心设计的C++试卷

航空工业出版社

普通高等教育“十一五”规划教材

C++程序设计教程

主 编 田秀霞 徐建平 彭 源

副主编 谢银勋 徐金鑫 马子孔 高振栋

航空工业出版社

北 京

内 容 提 要

C++程序设计课程是高等院校计算机及相关专业的重要基础课程,它是为满足计算机应用及软件开发领域对计算机应用人才的需要而设置的。通过本课程的学习,将为学习诸如软件工程、C#、基于C++的数据结构等重要课程打下坚实的基础。

本书以C++标准为蓝本,全面系统地讲述了C++语言的面向过程和面向对象的编程方法,内容涉及C++语言中的基本概念和基本知识(具体包括:C++编程环境概述、数据类型与表达式、C++语句的类型与语法、函数、数组与指针等),C++语言面向对象的基本特征(具体包括:类和对象、继承性、派生性、多态性和虚函数等),并引入了C++的许多新的语言特性(具体包括:命名空间、异常处理和模板等)。

本书语言简练、通俗易懂、结构新颖、条理清晰、重点突出、案例恰当、习题丰富,可供高等院校计算机应用及相关专业作为教材,也可供软件开发人员学习C++之用。

图书在版编目(CIP)数据

C++程序设计教程 / 田秀霞, 徐建平, 彭源主编. —北京: 航空工业出版社, 2010.9
ISBN 978-7-80243-441-7

I. C… II. ①田…②徐…③彭… III. C语言—程序设计—教材 IV. TP312

中国版本图书馆CIP数据核字(2010)第015310号

C++程序设计教程 C++ Chengxu Sheji Jiaocheng

航空工业出版社出版发行

(北京市安定门外小关东里14号 100029)

发行部电话: 010-64815615 010-64978486

北京市科星印刷有限责任公司印刷

全国各地新华书店经售

2010年9月第1版

2010年9月第1次印刷

开本: 787×1092

1/16

印张: 23.5

字数: 587千字

印数: 1—3000

定价: 39.80元



进入二十一世纪以来，全球软件产业总产值以年均 6% 以上的速度增长，其变革和技术发展给中国的软件行业带来了巨大的发展空间和发展契机。与此同时，软件和信息服务业正在以前所未有的速度向经济和社会生活的各个方面渗透。

伴随着中国软件行业的突飞猛进，该行业对人才的需求量也呈明显的上升趋势。软件企业急需吸纳优秀的软件人才，以增强企业在人才方面的竞争优势。

C++ 语言作为当今软件开发领域的主流程序设计语言，为此，很多学校都开设了 C++ 程序设计课程。目的是使学生进一步掌握使用高级程序设计语言开发应用程序的方法，并着重了解面向对象程序设计的基本概念与方法，同时为以后学习软件工程和基于 C++ 的数据结构等后续课程奠定程序设计基础。

本书特色

在本书编写过程中，我们以 C++ 标准为蓝本，全面系统地介绍了 C++ 语言面向过程和面向对象的编程方法，力求做到如下两点：

- ☞ **内容全面、结构完整、突出实用。**全书以 C++ 语言的使用规范为核心，以面向对象的程序设计思想为主线，对 C++ 语言进行了全面具体的介绍，并在内容翔实的基础上突出实用性，以反映软件领域的最新发展成果。
- ☞ **体例新颖、注重实效。**本书每章开头都有“引子”和“学习目标”，文中有大量的例题和“上机实训”，篇末有“本章小结”和“习题与上机练习”。从而使本书既便于学生系统地学习又便于随学随练，进而提高学生的实际编程技能。

本书内容

本书主要包括两大部分，第一部分主要介绍了 C++ 语言基础和 C++ 面向过程的编程知识，具体包括如下内容：C++ 及其编译环境，C++ 数据类型与表达式，C++ 程序设计基础，函数、数组、指针及其应用，自定义数据类型。

第二部分主要介绍了 C++ 面向对象的编程知识，具体包括：类和对象，继承与派生，多态性、虚函数与运算符重载，输入/输出流，命名空间与异常处理，模板的使用。

建议课时

本教程可以分两学期来讲，第一学期建议 68 学时，其中实验 28 学时；第二学期 68 学时，其中实验 24 学时。



本书配套资源

- ☞ 本书所有例子的源代码
- ☞ 部分课后习题的参考答案
- ☞ 教学课件

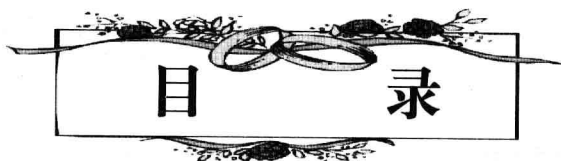
上述资源可到我们的网站 (<http://www.bjjqe.com>) 去下载。

分工与致谢

本书由田秀霞、徐建平、彭源任主编，谢银勋、徐金鑫、马子孔、高振栋任副主编。我们在本书编写过程中参考了大量的文献资料，在此，谨向参考过的中外文献的作者表示诚挚的谢意。此外，尽管我们在编写本书时已尽了最大的努力，但由于编者水平有限，书中个别地方难免会有疏漏之处，敬请广大读者批评指正。

编 者

2010 年 8 月



第 1 篇 C++语言基础与面向过程程序设计

第 1 章 C++及其开发环境概述	2
1.1 面向对象的程序设计概述	2
1.1.1 什么是面向对象的程序设计思想	2
1.1.2 面向过程与面向对象程序设计特点比较	4
1.1.3 两种程序设计方法对比分析	5
1.2 从 C 到 C++的演变	6
1.3 认识 C++程序及其开发环境	7
1.3.1 C++程序的开发过程	7
1.3.2 认识 C++程序	8
1.3.3 熟悉 C++开发环境——Visual C++ 6.0	10
上机实训	12
实训 1-1 编写并运行 C++程序	12
实训 1-2 熟悉 Visual C++工作区与工程的使用	15
本章小结	19
习题与上机练习	19
第 2 章 C++程序设计基础（上）	21
2.1 C++的数据类型	21
2.1.1 C++数据类型概述	22
2.1.2 关于数值的范围	24
2.1.3 数据类型的转换	25
2.1.4 sizeof 操作符	27
2.2 常量和变量	28
2.2.1 常量的类型与符号常量的声明	28
2.2.2 变量的命名、定义与声明	32
2.3 C++的运算符、表达式和优先级	34
2.3.1 算术运算符与算术表达式	35
2.3.2 赋值运算符与赋值表达式	37
2.3.3 关系运算符与关系表达式	38
2.3.4 逻辑运算符与逻辑表达式	39
2.3.5 条件运算符及其表达式	40



2.3.6 逗号运算符及其表达式	41
2.3.7 位运算符和位运算	42
2.3.8 运算符的优先级	43
上机实训	44
实训 2-1 输出多种立体的的体积	44
实训 2-2 判断三角形类型并求解其面积	47
本章小结	49
习题与上机练习	49
第 3 章 C++程序设计基础（下）	52
3.1 C++语句的类型	52
3.1.1 声明语句	52
3.1.2 执行语句	53
3.1.3 空语句	54
3.1.4 复合语句	54
3.2 数据的输入输出	54
3.2.1 输入语句	54
3.2.2 输出语句	55
3.2.3 输入/输出的格式控制	56
3.3 程序的流程控制	58
3.3.1 选择结构	58
3.3.2 循环结构	65
3.3.3 转向语句	70
3.4 预处理命令	73
3.4.1 宏定义	73
3.4.2 文件包含	74
3.4.3 条件编译	75
上机实训	76
实训 3-1 根据输入的年龄判断所属年龄阶段	76
实训 3-2 将输入数字转换为以汉字形式输出的货币金额	77
实训 3-3 输出杨辉三角	80
本章小结	82
习题与上机练习	83
第 4 章 函数的应用	87
4.1 函数声明	87
4.1.1 函数的定义	88
4.1.2 函数原型的声明	90
4.2 函数的参数与返回值	92
4.2.1 函数的参数	92
4.2.2 函数的返回值	94



4.3 函数调用	95
4.3.1 函数调用的一般形式	95
4.3.2 函数的嵌套调用	95
4.3.3 函数的递归调用	97
4.4 内联函数	99
4.5 函数重载	100
4.6 变量的存储属性	102
4.6.1 变量的作用域	102
4.6.2 变量的存储类别	105
4.7 内部函数和外部函数	108
上机实训	109
实训 4-1 编排日历	109
实训 4-2 求两点间距离	112
实训 4-3 汉诺塔问题求解	113
实训 4-4 用二分法迭代求解方程	114
本章小结	115
习题与上机练习	116
第 5 章 数组与字符串	119
5.1 认识数组	119
5.2 一维数组	119
5.2.1 一维数组的定义	120
5.2.2 一维数组的初始化	121
5.2.3 一维数组元素的引用	122
5.2.4 一维数组的应用	122
5.3 二维数组	125
5.3.1 二维数组的定义	125
5.3.2 二维数组的初始化	126
5.3.3 二维数组元素的引用	128
5.3.4 二维数组的应用	128
5.4 数组在函数中的应用	131
5.5 字符数组与字符串	132
5.5.1 字符数组的定义和初始化	132
5.5.2 字符数组的输入输出	133
5.5.3 字符串处理函数	134
5.6 字符串类和字符串变量	136
5.6.1 字符串变量	136
5.6.2 字符串数组	137
上机实训	138
实训 5-1 插入排序	138



实训 5-2 打印逆时针螺旋方阵	139
实训 5-3 找出最大字符串	140
本章小结	141
习题与上机练习	142
第 6 章 指针与引用	147
6.1 指针与指针变量	147
6.1.1 指针的概念	147
6.1.2 指针变量的定义及应用	148
6.1.3 指针运算	151
6.1.4 new 和 delete 运算符	153
6.2 指针与数组	154
6.2.1 指向一维数组的指针	154
6.2.2 二维数组的访问	156
6.3 指针与字符串	159
6.4 指针与函数	160
6.4.1 指针作函数参数	160
6.4.2 指向数组的指针作函数参数	162
6.4.3 返回指针的函数	163
6.4.4 指向函数的指针	164
6.5 指针数组和指向指针的指针	166
6.6 引用	169
6.6.1 变量的引用	169
6.6.2 引用作为函数的参数	170
上机实训	171
实训 6-1 降序输出数据	171
实训 6-2 矩阵转置	172
实训 6-3 输出成绩	174
本章小结	176
习题与上机练习	176
第 7 章 自定义数据类型	181
7.1 结构体	181
7.1.1 结构体变量的定义、初始化及使用	182
7.1.2 结构体数组的定义、初始化及使用	186
7.1.3 指向结构体变量的指针	189
7.1.4 结构体作为函数参数	191
7.2 共用体	192
7.2.1 共用体变量的定义与初始化	193
7.2.2 共用体变量的使用	194
7.3 枚举类型	195

7.3.1 枚举类型变量的定义与初始化	195
7.3.2 枚举类型变量的使用	195
7.4 typedef 声明	197
上机实训	198
实训 7-1 输出教师信息	198
实训 7-2 输出每种动物的信息	200
本章小结	202
习题与上机练习	203

第 2 篇 C++面向对象程序设计

第 8 章 类和对象	208
8.1 类	208
8.1.1 类的声明	208
8.1.2 类成员的访问属性	209
8.1.3 成员函数的定义	209
8.2 对象	211
8.2.1 对象的定义	211
8.2.2 对象成员的引用	213
8.2.3 类和对象的应用举例	213
8.3 构造函数	215
8.3.1 构造函数概述	215
8.3.2 带参数的构造函数	216
8.3.3 带默认参数的构造函数	217
8.3.4 带参数初始化表的构造函数	218
8.3.5 构造函数的重载	219
8.3.6 拷贝构造函数	221
8.4 析构函数	222
8.5 对象数组	223
8.5.1 对象数组的定义	223
8.5.2 对象数组的初始化	224
8.6 对象指针	225
8.6.1 对象指针的声明与使用	225
8.6.2 指向对象成员的指针	226
8.6.3 this 指针	227
8.7 静态成员	228
8.7.1 静态数据成员	228
8.7.2 静态成员函数	230



8.8 友元	233
8.8.1 友元函数	233
8.2.2 友元类	234
上机实训	235
实训 8-1 模拟存款与取款过程	235
本章小结	238
习题与上机练习	238
第9章 继承与派生	240
9.1 类的继承与派生概述	240
9.1.1 继承与派生的概念	240
9.1.2 派生类的声明	241
9.1.3 派生类的构成	242
9.1.4 派生类的继承方式	243
9.1.5 派生类的构造函数和析构函数	248
9.2 多重继承	254
9.2.1 多重继承的声明	254
9.2.2 多重继承派生类的构造函数	254
9.2.3 二义性问题	256
9.3 虚基类	260
9.3.1 虚基类的声明	260
9.3.2 虚基类的构造函数	261
上机实训	263
实训 9-1 通话计费	263
实训 9-2 工资管理	266
本章小结	271
习题与上机练习	271
第10章 多态性、虚函数与运算符重载	272
10.1 多态性的概念	272
10.2 虚函数	272
10.3 纯虚函数与抽象类	274
10.4 运算符重载	275
10.4.1 运算符重载的定义	275
10.4.2 运算符重载的形式	277
10.4.3 常用运算符的重载	281
10.5 实现类型转换	284
上机实训	286
实训 10-1 利用虚函数求不同图形的面积	286
实训 10-2 实现复数的简单运算	288
本章小结	290

习题与上机练习	290
第 11 章 输入/输出流	292
11.1 C++的输入/输出	292
11.1.1 流与缓冲区	292
11.1.2 输入/输出流类	292
11.2 标准输入输出流	294
11.2.1 标准输入流对象 cin	294
11.2.2 输入流的成员函数	295
11.2.3 标准输出流对象	299
11.2.4 输入/输出的格式控制	300
11.2.5 流运算符的重载	304
11.3 文件流	306
11.3.1 文件的特点	307
11.3.2 文件流类和流对象	307
11.3.3 文件的打开与关闭	308
11.3.4 文件的读写操作	310
11.3.5 文件指针的使用	313
11.3.6 二进制文件的应用实例（选读）	314
上机实训	318
实训 11-1 求解一元二次方程	318
实训 11-2 数据的写入与读出	319
本章小结	321
习题与上机练习	321
第 12 章 命名空间与异常处理	323
12.1 命名空间	323
12.1.1 命名空间的定义	325
12.1.2 命名空间成员的引用	328
12.2 异常处理	331
12.2.1 异常处理的实现	331
12.2.2 类的异常处理	337
12.2.3 异常规范	338
上机实训	340
实训 12-1 显示学生的个人信息和购书信息	340
实训 12-2 学生考勤情况统计	342
本章小结	347
习题与上机练习	347
第 13 章 模板的使用	350
13.1 模板概述	350
13.1.1 函数模板	351



13.1.2 类模板	351
13.2 函数模板	352
13.2.1 函数模板的定义	352
13.2.2 函数模板的应用	353
13.3 类模板	354
13.3.1 类模板的定义	354
13.2.2 类模板的应用	355
上机实训	357
实训 13-1 冒泡排序	357
实训 13-2 选择不同的排序方法排序	360
本章小结	362
习题与上机练习	362

第 1 篇 C++语言基础与 面向过程程序设计

- C++及其开发环境概述
- C++程序设计基础（上）
- C++程序设计基础（下）
- 函数的应用
- 数组与字符串
- 指针与引用
- 自定义数据类型

第 1 章 C++ 及其开发环境概述

【引 子】

C++ 是一款非常优秀的程序设计语言，它不仅继承了 C 语言的许多优点，而且增加了面向对象的高级特性，因此它逐渐成为最重要的程序设计语言之一。在本章中，我们首先介绍面向对象的程序设计思想，以及面向过程和面向对象程序设计的区别，并通过实例进行说明。之后将通过认识一个简单的 C++ 程序，以及它的编写、编译、连接和运行，从而帮助读者认识 C++ 程序及其开发环境。

【本章学习目标】

- ◆ 了解什么是面向对象的程序设计
- ◆ 了解从 C 到 C++ 的演变过程
- ◆ 掌握 C++ 的代码构成、书写形式和编写、编译、运行方法
- ◆ 熟悉 C++ 的集成开发环境——Visual C++ 6.0

1.1 面向对象的程序设计概述

与 C 相比，C++ 的最大特点是它很好地支持面向对象的程序设计。那么，什么是面向对象的程序设计呢？

1.1.1 什么是面向对象的程序设计思想

早期的程序设计被称为结构化程序设计或面向过程的设计。当我们编写一个程序时，主要精力都花在了数据结构设计和程序代码（各种函数或过程）的编写上。因此，结构化程序设计又称以算法为核心的程序设计。

例如，窗口是绝大部分应用程序都要用到的。如果使用传统的程序设计方法，单单开发窗口界面就要耗费很长的时间。虽然很多程序语言（如 C）也提供了众多标准化函数，但由于函数的层次较低，我们依然需要编写大量的程序代码。此外，如果我们以后又需要开发一个新的应用程序，虽然一些代码可以拿过来使用，但仍需要深入到某些子程序内部去修改数据结构或程序代码。

那么，有没有一种更好的方法来克服这些问题呢？这便导致了面向对象的程序设计思想的诞生。下面我们将结合 C++ 具体介绍这种程序设计思想。

通俗地讲，所谓面向对象的程序设计就是指以对象为编程核心的程序设计思想。在这种程序开发模式下，我们的关注点主要是对象，而不再主要是具体的算法设计。我们可以通过对象的数据接口（在 C++ 中称为数据成员）向对象传递数据和获取数据，通过对象的操作接

口（在 C++ 中称为成员函数）来实现对数据的操作。至于对象内部如何对数据进行处理，以及各种操作的具体实现，我们将不再关心。

但是，话虽这样说，面向对象的程序设计内涵是非常丰富的。因此，它涉及的概念非常多。下面我们首先介绍几个面向对象程序设计所涉及的最基本概念，至于更进一步的知识，将在本书的第二部分详细阐述。

1. 对象

在现实世界中，对象可以是我们要研究的任何事物，它既可以是有形的具体存在的事物，如一辆汽车、一本书、一个学生；也可以是无形或抽象的事件，如一次演出、一个规划、一次出差等。对象既可以很简单，也可以很复杂，复杂的对象可以由多个简单的对象组成。

一般来说，现实世界中的对象既有反映其特征的静态的属性（或称状态），又有动态的行为（或称操作、功能、方法等）。例如，假定把一个人作为一个对象，他应该有姓名、性别、年龄、身高、体重等属性，以及走路、吃饭、睡觉、学习等行为。

面向对象程序设计中的对象实际上就模拟了上述原理。人们将用于描述对象属性的数据，以及对这些数据施加的一组操作封装起来，这便是对象。编程时，我们只需通过对象的数据接口和操作接口，便可以让对象为我们所用了。

2. 类

类是具有相同属性和行为的一组对象的集合，它为属于该类的全部对象提供了统一的抽象描述。或者说，类是对象的抽象，对象是类的实例化。例如，我们可以将“人”看成类，而将张三、李四等具体的人看成对象。

在 C++ 中，我们必须利用类来创建对象，而类既可以自己定义，也可以使用系统提供的或他人编写的。此外，我们可以利用一个类创建多个对象，并可以分别对这些对象进行不同的操作。

3. 封装

如前所述，对于类及其实例化的对象而言，相关的数据和操作代码都被放在了类及对象内部，使用者无法了解（当然也不必了解）其内部细节，这便是封装。正如一台洗衣机一样，我们看到的是它的颜色、形状、尺寸等外观，能操作的是它的按钮。按下某个按钮后，至于洗衣机内部是如何工作的，我们无法也没有必要了解了。

4. 继承

在定义和实现一个类的时候，可以在一个与其内容基本相同的类的基础之上来进行，把这个已经存在的类所定义的内容作为自己的内容，并加入若干新的内容即可。这就是面向对象中的继承机制。如“日语书”继承了“书”的基本特征，又增加了新的特征。“书”是父类，也叫基类；“日语书”是从“书”派生出来的，称为子类或派生类。

5. 多态

多态是指：对于不同的对象，当它们收到同一消息时会调用不同的方法（程序），从而执行不同的操作。在 C++ 中，多态是通过函数重载、运算符重载和虚函数实现的。所谓函数重载是指：在程序中定义若干名字相同的函数，但其参数的类型或个数不同。如此一来，调



用函数时通过指定不同类型或个数的参数，便可以调用不同的函数。

1.1.2 面向过程与面向对象程序设计特点比较

目前，几乎所有的编程语言都引入了面向对象的程序开发机制，这绝非偶然。其根本原因在于面向过程的程序开发方法因其固有的缺陷，已远远不能满足当今软件开发的需要。下面就让我们对两者进行一个简单的比较。

1. 面向过程的程序设计方法的缺陷

对于面向过程的程序设计而言，其本质是程序功能的逐层分解，然后可通过设计合理的数据结构，并借助函数和具体的语句来实现这些功能。正是这种机制导致其缺陷众多，具体表现如下。

(1) 程序的开发效率低下

由于开发时需要编写大量的程序代码，并精心设计每段程序的算法，因此，程序开发的工作量巨大，进而导致程序的开发效率低下。尤其是当程序规模较大，数据较多，函数关系非常复杂的时候，编写程序的过程将会变得异常困难。

(2) 程序的可重用性差

由于程序和数据密切相关，因此，当我们需要使用以前编写的某些子程序时，需要精心检查和修改程序，否则很可能会出错。因此，这种编程方式下程序的可重用性很差。

(3) 程序的维护成本高

由于程序代码众多，算法复杂，因此，当我们需要修改程序的某项功能时，很可能会牵一发而动全身，从而导致程序的维护成本高昂。

(4) 无法适应信息的快速膨胀和多样化

随着计算机应用广度和深度的不断扩展，计算机处理的数据早已不再仅仅局限在简单的数字和字符，而是包括了声音、视频、图形、图像等各种格式的数据，并且数据量也在快速膨胀。面对这种情况，传统的面向过程的程序设计方法已无能为力。

2. 面向对象的程序设计方法的优势

与面向过程的程序设计方法相比，前者的劣势正是面向对象程序设计方法的优势，具体包括以下几点。

(1) 程序的开发效率高

由于这种编程方法以对象操作为核心，因此，很多情况下只需围绕对象编写少量程序代码即可实现复杂的功能，从而大大提高了程序的开发效率，并且可以更好地支持大型应用程序的开发。

(2) 程序代码的可重用性高

由于类具有良好的封装性和继承性，因此，很多程序模块都可以在各种应用程序中直接使用，或者仅需稍加修改即可使用，故而程序代码具有较高的可重用性。

(3) 程序易于维护

由于使用这种编程方法开发应用程序时编写的程序代码较少，且类具有良好的封装性，因此，使用这种方法开发的程序自然易于维护。