

## <<高级程序设计技术>>

### 图书基本信息

书名：<<高级程序设计技术>>

13位ISBN编号：9787115214560

10位ISBN编号：7115214565

出版时间：2009-11

出版时间：人民邮电出版社

作者：曾凡仔 等编著

页数：328

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## &lt;&lt;高级程序设计技术&gt;&gt;

## 前言

C++语言是目前使用最为广泛的一种高效程序设计语言，它既可以进行过程化的程序设计，也可以用于面向对象的程序设计。

C++是从C语言发展演变而来的，是C语言的超集。

它实现了类的封装、数据隐藏、继承及多态，使得其代码容易维护及高度可重用。

本书作为C++语言的入门教材，不仅详细地介绍了C++语言本身，还深入地阐述了面向对象的程序设计方法。

本书的主要特点是语言流畅，简洁易懂，例题丰富，实用性强。

这使读者不仅可以学会一门程序设计语言，还能初步掌握面向对象的程序设计方法。

其中，丰富的例题使初学者可以在学习的同时就开始积累初步的编程经验，以尽快达到学以致用的目的。

本书共计8章。

第1章C++语言概述，主要介绍C++的发展历史，面向对象的程序设计概念，C++的词法与规则，C++程序的结构与实现；数据类型，常量、变量，运算符与表达式，流控制，数据的输入/输出；C++语句，顺序、分支和循环程序设计。

第2章类和数据抽象，主要介绍类、对象，对象的初始化，构造函数与析构函数，堆与拷贝构造函数，静态成员和静态成员函数，友元函数和友元类，this指针等。

第3章继承和派生，主要介绍基类和派生类，单继承、多继承和虚基类。

第4章多态性与虚函数，主要介绍模板的概念，函数模板和类模板，运算符重载，普通成员函数重载，构造函数重载，派生类指针，虚函数，纯虚函数和抽象类等。

第5章C++流，主要介绍I/O标准流类，键盘输入、屏幕输出，磁盘文件的输入和输出，字符串流等内容。

第6章线性表，主要介绍线性表的逻辑结构，线性表的顺序存储及运算实现，线性表的链式存储和运算实现及顺序表和链表的比较等内容。

第7章查找，主要介绍查找的基本概念，静态查找表，动态查找表，哈希表查找等内容。

第8章排序，主要介绍排序的基本概念，插入排序，交换排序，选择排序，二路归并排序，基数排序和外排序等内容。

附录中给出了Linux环境下C++编译系统提供的库函数和类库，以方便读者查阅。

本书中所有例题都在Linux环境下C++编译系统中运行通过，在其他版本的编译系统中一般也都可以运行。

本书为高职高专院校的C++程序设计教材，建议教授课时为45课时，上机实践课时为45课时，课程设计课时为16课时。

各院校可根据教学实际情况适当增删。

本书也可作为大中专院校的程序设计课程教材和各类培训机构培训教材，还可供从事计算机应用的工程技术人员参考。

本书在编写过程中，编者参阅了许多C++的参考书和有关资料，并参阅了一些翻译的书籍，现谨向这些书的作者和译者表示衷心的感谢。

由于编者水平所限，书中难免有不妥或错误之处，欢迎广大读者指正。

## <<高级程序设计技术>>

### 内容概要

本书全面系统地阐述了C++语言的基本概念、语法和面向对象的编程方法；对C++语言面向对象的基本特征：类和对象、继承性、派生类、多态性和虚函数等内容作了详尽的介绍；从软件开发的实际需要出发，按照面向对象的程序设计思想，详细地介绍了线性表、查找、排序等数据结构及其算法实现。

本书列举了丰富的例题，每章后面备有形式多样的练习题。

在内容安排上循序渐进、深入浅出，力求通俗易懂、突出重点、侧重应用。

本书不仅可作为高职高专院校和培训机构C++语言程序设计的教材，也可作为自学C++语言的指导用书和计算机工程技术人员的参考书。

## &lt;&lt;高级程序设计技术&gt;&gt;

## 书籍目录

第1章 C++语言概述	1.1 C++语言的起源与特点	1.1.1 从C到C++	1.1.2 C++与C的关系
	1.1.3 C++面向对象的特性	1.2 C++语言的基本符号与词法	1.2.1 C++的字符集
	1.2.2 数据类型概述	1.2.3 常量	1.2.4 变量
	1.2.5 运算符	1.2.6 表达式	1.2.7 数据类型转换
1.3 C++语言程序的结构	1.3.1 顺序结构	1.3.2 选择结构	1.3.3 循环结构
1.3.4 转移语句	1.4 C++语言程序的编辑及运行	1.4.1 Linux程序设计基础知识	1.4.2 Linux下C++语言编程环境概述
1.4.3 Linux下C++语言编码的风格	习题	第2章 类和数据抽象	2.1 类的定义
2.1.1 类的定义	2.1.2 类的成员函数	2.1.3 类和结构	2.2 对象的创建与成员引用
2.2.1 对象的说明	2.2.2 对象的生存期	2.2.3 类作用域	2.2.4 引用
2.2.5 常类型	2.3 构造函数与析构函数	2.3.1 构造函数	2.3.2 析构函数
2.3.3 缺省构造函数和缺省析构函数	2.3.4 带参数的构造函数	2.3.5 内联函数和外联函数	2.3.6 堆对象与拷贝构造函数
2.3.7 局部类和嵌套类	2.4 友元函数与友元类	2.4.1 友元函数的说明	2.4.2 友元函数的使用
2.4.3 友元类	2.5 静态成员	2.5.1 静态数据成员	2.5.2 静态成员函数
2.6 this指针	习题一	习题二	第3章 继承和派生
3.1 基类和派生类	3.1.1 派生类的定义格式	3.1.2 派生类的3种继承方式	3.1.3 访问控制
3.1.4 基类和派生类的关系	3.2 继承方式	3.2.1 单继承	3.2.2 多继承
3.2.3 虚基类	3.3 派生与继承应用实例	3.3.1 问题描述	3.3.2 算法分析
3.3.3 数据说明	3.3.4 功能说明：定义父类和相关的子类	3.3.5 参考程序：“院校管理系统”程序实例	习题
第4章 多态性与虚函数	4.1 重载	4.1.1 运算符重载	4.1.2 普通成员函数重载
4.1.3 构造函数重载	4.1.4 派生类指针	4.1.5 模板	4.2 虚函数
4.2.1 静态联编与动态联编	4.2.2 虚函数的概念	4.2.3 动态联编与虚函数	4.2.4 虚函数的限制
4.2.5 虚函数与重载函数的比较	4.3 纯虚函数和抽象类	4.3.1 纯虚函数	4.3.2 抽象类
4.3.3 虚析构函数	习题一	习题二	习题三
第5章 C++流	第6章 线性表	第7章 查找	第8章 排序
参考文献			

## 章节摘录

1.1.3C++面向对象的特性 1.C++支持数据封装 支持数据封装就是支持数据抽象。

在C++中，类是支持数据封装的工具，对象则是数据封装的实现。

面向过程的程序设计方法与面向对象的程序设计方法对待数据和函数关系是不同的，在面向对象的程序设计中，将数据和对该数据进行合法操作的函数封装在一起作为一个类的定义，数据将被隐藏在封装体中，该封装体通过操作接口与外界交换信息。

对象被说明为具有一个给定类的变量，类似于C语言中的结构，在C语言中可以定义结构，但这种结构包含数据，而不包含函数。

C++中的类是数据和函数的封装体。

在C++中，结构可作为一种特殊的类，它虽然可以包含函数，但是它没有私有或保护的成员。

2.C++类中包含私有、公有和保护成员 C++类中可定义3种不同访控制权限的成员。

一种是私有（priVate）成员，只有在类中说明的函数才能访问该类的私有成员，在该类外的函数不可以访问私有成员；另一种是公有（public）成员，类外面也可访问公有成员，成为该类的接口；还有一种是保护（protected）成员，这种成员只有该类的派生类可以访问，其余的在这个类外不能访问。

3.C++中通过发送消息来处理对象 C++中是通过向对象发送消息来处理对象的，每个对象根据所接收到的消息的性质来决定需要采取的行动，以响应这个消息。

响应这些消息是一系列的方法，方法是在类定义中使用函数来定义的，使用一种类似于函数调用的机制把消息发送到一个对象上。

4.C++中允许友元破坏封装性 类中的私有成员一般是不允许该类外的任何函数访问的，但是友元可打破这条禁令，它可以访问该类的私有成员（包含数据成员和成员函数）。

友元可以是在类外定义的函数，也可以是在类外定义的整体类，前者称为友元函数，后者称为友元类。

友元打破了类的封装性，它是C++另一个面向对象的重要性。

.....

#### 版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>