

## <<C++面向对象程序设计>>

### 图书基本信息

书名：<<C++面向对象程序设计>>

13位ISBN编号：9787302244653

10位ISBN编号：7302244650

出版时间：2011-3

出版时间：清华大学出版社

作者：龚晓庆 等编著

页数：350

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>

## <<C++面向对象程序设计>>

### 内容概要

本书讲解如何用c++程序设计语言编写面向对象程序。

本书以面向对象概念为主线索，内容由浅入深，主要包括面向对象程序设计方法、c++程序结构、c++基本语法、函数、名字空间、类和对象、构造函数和析构函数、运算符重载、包含与继承、虚函数与多态性、模板、异常处理、c++标准i/o流库和标准模板库。

本书强调c++语言的实用性，从c++语言的语法、语义和语用3个层面着手，除了详细介绍c++的各种语法概念及其语义之外，更着重其运用知识：在特定情况下应该使用何种语法结构，用它们来解决什么样的问题。

本书穿插介绍了一些实用的编程经验，例如程序的多文件组织、变量命名、函数的设计、类的设计、错误处理技术等。

本书可作为高等院校计算机专业的c++程序设计教材。

# <<C++面向对象程序设计>>

## 书籍目录

### 第1章 面向对象基础

- 1.1 程序设计范型
- 1.2 程序设计与抽象的过程
- 1.3 面向对象的基本概念
- 1.5 面向对象方法发展简史
- 1.6 小结
- 1.7 习题

### 第2章 c++语言概览

- 2.1 c++语言的特点
- 2.2 第一个c++程序
- 2.3 c++与过程式程序设计
- 2.4 c++与基于对象的程序设计
- 2.5 c++与面向对象程序设计
- 2.6 c++与泛型程序设计
- 2.7 异常处理
- 2.8 c++程序的结构和编译
- 2.9 c++开发工具
- 2.10 小结
- 2.11 习题

### 第3章 c++基本语法

- 3.1 数据类型
- 3.2 常量和变量
- 3.3 运算符和表达式
- 3.4 语句
- 3.5 小结
- 3.6 习题

### 第4章 函数

- 4.1 函数的基本概念
- 4.2 inline函数
- 4.3 缺省函数参数
- 4.4 重载函数
- 4.5 链接指示：extern “c”
- 4.6 函数指针
- 4.7 名字空间
- 4.8 作用域和存储期
- 4.9 程序代码组织
- 4.10 设计高质量的函数
- 4.11 小结
- 4.12 习题

### 第5章 类和对象

- 5.1 类的定义
- 5.2 对象的创建与销毁
- 5.3 类设计的例子
- 5.4 const成员
- 5.5 static成员

## <<C++面向对象程序设计>>

- 5.6 指向成员的指针
- 5.7 类代码的组织
- 5.8 小结
- 5.9 习题
- 第6章 运算符重载
  - 6.1 运算符重载的基本语法
  - 6.2 常用运算符的重载
  - 6.3、重载赋值运算符
  - 6.4 重载下标运算符
  - 6.5 用户定义的类型转换
  - 6.6 小结
  - 6.7 习题
- 第7章 包含与继承
  - 7.1 包含——复用类的实现
  - 7.2 继承——复用类的接口
  - 7.3 多重继承
  - 7.4 包含与继承的选择
  - 7.5 小结
  - 7.6 习题
- 第8章 虚函数与多态性
  - 8.1 向上类型转换
  - 8.2 虚函数
  - 8.3 抽象类
  - 8.4 rtti
  - 8.5 类层次设计的例子
  - 8.6 小结
  - 8.7 习题
- 第9章 模板
  - 9.1 函数模板
  - 9.2 类模板
  - 9.3 模板的编译模式
  - 9.4 模板和代码复用
  - 9.5 小结
  - 9.6 习题
- 第10章 异常处理
  - 10.1 抛出异常
  - 10.2 try块
  - 10.3 捕获异常
  - 10.4 异常规范
  - 10.5 异常类层次
  - 10.6 异常和错误处理技术
  - 10.7 小结
  - 10.8 习题
- 第11章 输入/输出流
  - 11.1 输入/输出流类层次
  - 11.2 终端输入/输出
  - 11.3 文件输入/输出

## <<C++面向对象程序设计>>

11.4 字符串流

11.5 小结

11.6 习题

第12章 标准模板库

12.1 stl概览

12.2 顺序容器

12.3 迭代器

12.4 关联容器

12.5 泛型算法

12.6 小结

12.7 习题

附录a

参考文献

## &lt;&lt;C++面向对象程序设计&gt;&gt;

## 章节摘录

本章将介绍面向对象程序设计的基本概念。

对许多人来说，如果不了解面向对象的背景知识而直接进入面向对象程序设计，可能会有些困难。

这里预先介绍面向对象技术的一些基础知识，作为一个简要的参考。

有些读者可能要看到具体的语言和程序结构之后才能了解其整体概念，那么可以跳过本章，不会妨碍后续章节C++语言的学习。

但是最终回过头来补充本章内容，对于理解对象的重要性和面向对象程序设计的必要性也有助益。

1.1 程序设计范型 程序主要由算法和数据两个方面组成。

在计算机的发展史上，程序的这两个主要方面一直保持不变，但它们之间的关系却在不断地发展演化，形成了所谓的程序设计方法，也被称作程序设计范型（paradigm）。

典型的程序设计范型包括：过程式程序设计、基于对象的程序设计、面向对象的程序设计和泛型程序设计。

过程式程序设计是最传统、使用最久的方法。

在过程式程序设计方法中，问题分解是控制复杂性的主要手段，一个问题可以由一组算法来建模。

对一个要解决的问题进行自上而下的逐级分解，得到一组子问题；再利用子过程来分别解决这些子问题，最终通过主程序中对子过程的调用实现整个问题的解。

程序处理的数据被独立存储起来，各个子过程可以在全局位置访问这些数据，或者将数据传递给过程以便其访问。

著名的过程式程序设计语言有Fortran、C、Pascal等。

20世纪70年代，程序设计的焦点从过程式程序设计转移到了抽象数据类型（Abstract Data Type，ADT）的设计上，现在称之为基于对象的程序设计。

在基于对象的程序设计方法中，通过一组数据抽象来对问题建模，这些抽象被称为类。

与类相关的算法被称为该类的公有接口，数据以私有的形式被存储在各个对象中，对数据的访问与一般的程序代码隔离开来。

系统则由类的对象实例之间的相互作用表现出来。

Ada是支持抽象数据类型的代表语言之一。

&hellip;&hellip;

## <<C++面向对象程序设计>>

### 编辑推荐

《C++面向对象程序设计》强调C++语言的实用性；像学习自然语言一样，可以从语法、语义和语用这三个层面来学习和理解C++语言。

除了详细解释C++的语法概念及其语义之外，更着重于C++的语用知识：在特定情况下应该使用何种语法结构，用它们来解决什么样的程序设计问题。

在各章的习题部分还增加了软件公司的C++面试题，希望能够让读者对C++在实际中的应用有更加广泛和深入的了解。

## <<C++面向对象程序设计>>

### 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>