

<<面向对象程序设计高级教程>>

图书基本信息

书名：<<面向对象程序设计高级教程>>

13位ISBN编号：9787040079227

10位ISBN编号：7040079224

出版时间：2002-5

出版时间：蓝色畅想图书发行有限公司

作者：陈奇

页数：340

字数：530000

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>

<<面向对象程序设计高级教程>>

前言

怎样学好一种语言？

怎样学会用正确的方法用这种语言编程？

这恐怕是所有渴望成为一个优秀软件工程师的人都必须面对的问题。

古人云：学而不思则罔，思而不学则怠。

这是一个很好的经验教训，不过还应该再补充一点：实践。

学好语言最重要的法宝是编程，再编程。

接触过不少同学，他们非常热爱计算机，非常渴望成为一个卓越的程序员，他们也很努力，但效果却不如人意，有些人甚至走了很长的弯路。

自己当初也经历过种种曲折，很希望做点什么对这些同学有所助益。

本书是讲述面向对象程序设计方法与C++的，但我们不希望把它仅仅写成一种语言教材。

学好语言不等于学好了编程，而不以编程为目的的语言学习也是不可能真正学好语言的。

本书在内容取舍、教材结构及习题设置方面都力求体现这一点。

20世纪90年代初以前，我一直是用c语言写软件，用Pascal讲课。

当时我们开了一门叫做高级程序设计的课程，力图告诉学生怎样才算是好程序，怎样才是一个高的编程境界。

多年后，一些学生回忆起来，仍觉得获益良多。

通过那门课的教学，我们（我和我的学生们）明白了一个道理，编程就像武侠小说里的练武，光是一招一式是不够的，要成为一个武林高手，就要掌握关键，就要追求那种境界。

当时，我们在课上读了不少非常优美的程序，也写了一些所谓的大作业（一千行开外，四、五千行以内），好像也挺优美的。

但一到写实际的程序（一般在数万行以上），最终总是写得很蹩脚。

当时用的是结构化编程，我自己的体会是很有道理的，但总是写不好。

一方面得承认自己练得不够纯熟，一方面好像总觉得结构化程序设计本身是一套境界不够高的“拳”，坦率地说，我一直没有在用c语言编写大程序上体会到多少内在的乐趣。

20世纪80年代末开始，我们接触到了面向对象程序设计，进而又接触了C++，当我们开始尝试用C++以面向对象方式编程时，各种条件都还比较差，国内教材几乎没有，当时流行的C++编译器都还只有C版本。

但即使在那样的条件下，当在写一个C++的软件系统时，我第一次在复杂程序组织方面体会到一种令人惊奇的美妙感觉。

很久以后，我才逐渐明白了面向对象思想巨大的魅力所在，这是一种令人振奋的体验。

面向对象真的那么好吗？

国外有人说，C++的好处你得写五万行以上的程序才能体会到。

这也许说得有些过，但确实，结构化程序设计和面向对象程序设计，都是程序越来越复杂的产物。

在本书中，我们给出了一个较大的程序实例，展示了采用结构化的C与面向对象的C++不同的实现方案。

但限于篇幅，我们仍没有机会接触真正复杂的程序，希望读者能多加一些想像去体会面向对象在复杂程序组织中的意义。

本书不打算写成一本书，所以读者不会在其中查到C++的全部细节。

我们认为目前主要的C++编程环境提供的联机帮助远比课本要好得多。

本书的目的是想帮助读者掌握面向对象编程与C++的关键。

以前接触过一些“精通”c++的学生，令人惊奇的是，他们对最关键、最重要的C++基本成分居然不得要领，甚至全然不知，而且更无法想像的是，这种情况还比较普遍，我们希望在“学”这一环节上，能给读者正确的引导。

<<面向对象程序设计高级教程>>

内容概要

本书针对学过一门结构化语言的读者，以C++为背景语言，全面介绍了面向对象程序设计的一些概念和方法。

主要内容包括：结构化程序设计，C语言回顾，面向对象程序设计基础，C++语言基础，面向对象的分析和设计，C++深入论题，标准模板库STL，Windows编程和MFC，一个综合实例研究。

全书系统地介绍了C++的要点与关键，深入分析了C++中的难点，并对重要的实现机制进行了必要的讨论，力求帮助读者在全面掌握面向对象程序设计方法与C++编程精髓的基础上，进入一种良好的编程境界。

本书强调实例分析与概念理论相结合，通过一个贯穿全书的实例——图书馆管理系统，比较完整地展示了采用结构化编程语言C与面向对象编程语言C++的不同实现方案，具有较强的实用性。

本书可作为高等学校计算机或相关专业的教材或参考书，也可供对计算机有较高要求专业的研究生使用。

对于希望深入掌握面向对象程序设计方法与C++编程技术的读者，本书也不啻是一本较好的参考书。

<<面向对象程序设计高级教程>>

书籍目录

第一章 结构化程序设计 1.1 软件和编程 1.2 结构化程序设计 1.3 层次树状的结构 1.4 模块化 1.5 自顶向下方法 1.6 软件开发的瀑布模型 习题1第二章 C语言回顾 2.1 C语言基础 2.1.1 编码 2.1.2 类型 2.1.3 三种控制结构 2.1.4 程序的结构 2.1.5 程序运行时的内存占用 2.1.6 文件 2.2 C语言中的指针 2.2.1 指针基础 2.2.2 指针和数组 2.2.3 指针的指针 2.2.4 更为复杂的指针 2.3 程序设计风格基础 2.3.1 程序的清晰性 2.3.2 程序的坚固性 2.3.3 程序的通用性 2.3.4 程序的交互性 2.4 用C语言实现的简单图书馆管理系统 习题2第三章面向对象程序设计基础 3.1面向对象的由来和发展 3.2类和对象 3.3面向对象中的抽象 3.4继承性和多态性 3.5面向对象方法和原型技术 习题3第四章C++语言基础 4.1 C++的发展 4.2 更好的C 4.2.1 简洁的单行注释 4.2.2 严格的参数检查 4.2.3 引用 4.2.4 灵活的局部变量说明 4.2.5 函数的缺省值 4.2.6 内联函数 4.2.7 常量修饰 4.2.8 空间申请和释放 4.3 数据抽象和封装 4.3.1 类的引入 4.3.2 类和对象 4.3.3 成员函数和this指针 4.3.4 构造和析构 4.3.5 常量成员函数 4.3.6 友元 4.4 继承 4.5 多态性 4.5.1 重载 4.5.2 虚函数 4.5.3 纯虚函数与抽象类 4.6 文件和流 4.6.1 文本流的操作 4.6.2 二进制流的操作 习题4第五章 UML与面向对象的分析与设计 5.1统一建模语言UML概述 5.1.1 UML的产生和成长 5.1.2 UML的内容 5.1.3 UML的应用领域 5.2 统一建模语言UML的静态建模机制 5.2.1 用例图 5.2.2 类图 5.2.3 组件图和配置图 5.3 统一建模语言UML的动态建模机制 5.4 使用UML的过程 5.5 UML的应用实例：一个图书馆信息系统第六章 C++深入论题第七章 标准模板库STL第八章 Windows编程和MFC第九章 综合实例：图书馆管理系统

<<面向对象程序设计高级教程>>

章节摘录

插图：程序设计主流已经从结构化程序设计过渡到了面向对象程序设计。

有人错误地以为这两者是完全对立的，其实不然，面向对象程序无论从发展的历史上看还是从其内在的意义上看，都是结构化程序设计的发展。

在一定意义上讲，可以认为面向对象程序设计更彻底地实现了结构化程序设计的理想。

为了更好地讨论面向对象程序设计，我们有必要首先来回顾一下结构化程序。

与此同时，目前面向对象程序设计的主流语言C++又恰恰是从结构化程序设计的主流语言C发展过来的，为后面讲述的方便，我们也一并回顾一下C语言中的难点、要点或一些容易忽略而又很有意义的问题。

已经熟悉这些内容的读者可以跳过本章。

1-1软件和编程计算机是20世纪人类伟大的成就之一。

计算机在诞生以来的50多年中的发展速度是令人震惊的，不仅在于其迅速提高的速度和容量，更在于其应用的广阔性。

计算机为什么会有如此超乎人们想像的今天呢？

一个重要的奥秘就是“软件”。

什么是软件？

认真一想一定会发现有些难以回答，软件是程序、工具还是一类机器？

或者是机器中的一种组成部分？

把软件看作是一类机器或机器部件是有道理的，但它和传统机器的最大区别就在于它的“软”，即容易改变。

计算机的一个基本设计就是将“机器”分成了两部分：一部分是具有普遍性的支撑平台，人们抽象出了基本的逻辑结构和物理结构，即计算机的硬件；另一部分是可以随时根据用户的需要定制和改变的，即软件。

制造和出售硬件的厂商在某种程度上不用关心他们制造的“机器”将会用来干什么，但在这种机器上附着的软件为它提供了无限的可能。

<<面向对象程序设计高级教程>>

编辑推荐

《面向对象程序设计高级教程》是由高等教育出版社出版的。

<<面向对象程序设计高级教程>>

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>