

<<易学C++>>

图书基本信息

书名：<<易学C++>>

13位ISBN编号：9787115177421

10位ISBN编号：7115177422

出版时间：2008-6

出版时间：人民邮电出版社

作者：潘嘉杰

页数：356

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>

## <<易学C++>>

### 内容概要

本书是为C++程序设计学习者量身订做的辅导书。

全书分为3篇。

第一篇介绍了面向过程的程序设计，主要有基本语句、语法基础、函数机制和数据类型等内容。

第二篇介绍了一些实用编程技巧，内容包括阅读代码、调试程序和简单的编程思想。

第三篇介绍了面向对象的程序设计，主要有类和对象、对象生灭、友元、继承等内容。

书中常以形象的比喻来解释程序设计中的概念，通俗易懂，令读者印象深刻，更快地进入C++程序设计的大门。

本书的内容涵盖了绝大部分常用的C++知识，可以作为大学计算机专业或非计算机专业的程序设计入门教材，也可供计算机爱好者自学使用。

## 书籍目录

第一篇 过程化的程序设计第1章 良好的学习开端1.1 软件与程序1.2 程序设计要做什么1.3 选好一种语言1.4 C++能够做些什么1.5 C语言、C++和Visual C++的关系1.6 学习程序设计的方法和必要准备1.7 总结第2章 Hello, World2.1 如何创建一个示例程序2.2 创建自己的Hello, World2.3 C++语言的输出与输入2.4 方法指导2.5 习题第3章 各种各样的“箱子”——变量3.1 会变的“箱子”——定义变量3.1.1 C++数据类型3.1.2 变量名3.1.3 变量的初始化3.2 常用的基本数据类型3.2.1 整型(Integer) 3.2.2 实型(Real) 3.2.3 字符型(Character) 3.2.4 布尔型(Boolean) 3.3 不会变的“箱子”——定义常量3.4 C++算术表达式3.4.1 赋值3.4.2 除、整除和取余3.5 “箱子”的转换——数据类型转换3.5.1 隐式转换3.5.2 显式转换3.6 方法指导3.7 习题第4章 要走哪条路——条件语句4.1 如果……4.1.1 条件——关系运算4.1.2 条件——逻辑运算4.1.3 &&和||的妙用4.2 否则……4.2.1 如果与否则4.2.2 如果里的如果——if的嵌套4.2.3 找朋友4.3 爱判断的问号4.4 切换的开关4.4.1 多路开关——switch4.4.2 巧用switch4.5 方法指导4.6 习题第5章 有个圈儿的程序——循环语句5.1 程序赛车5.1.1 循环语句for5.1.2 加加和减减5.1.3 巧用for5.2 退出比赛和进维修站5.2.1 退出比赛——break5.2.2 进维修站——continue5.3 圈圈里的圈圈5.3.1 C++循环的嵌套5.3.2 怎么让输出的东西更好看5.4 While循环5.4.1 当型循环5.4.2 导火索——do5.5 方法指导5.6 习题第6章 好用的“工具”——函数6.1 简单的“工具”——函数6.1.1 “工具”的说明书6.1.2 如何使用系统造好的“工具”6.2 打造自己的“工具”6.2.1 C++函数的声明6.2.2 函数的定义6.2.3 函数是如何运行的6.2.4 返回语句——return6.2.5 关于主函数6.2.6 同名同姓——参数定义6.2.7 函数存在的意义6.3 多功能“开瓶器”——函数重载6.4 自动的“工具”6.4.1 默认参数6.4.2 定义默认参数的顺序6.4.3 默认参数和重载函数的混淆6.5 给变量和参数起个“绰号”——引用6.5.1 引用的声明6.5.2 用引用传递参数6.6 \*函数里的函数——递归6.7 方法指导6.8 习题第7章 好大的“仓库”——数组7.1 让计算机处理更多数据——使用数组7.1.1 C++中数组的声明7.1.2 数组的操作7.1.3 数组的初始化7.1.4 省略数组大小7.2 仓库是怎样造成的7.2.1 内存和地址7.2.2 C++数组在内存中的存储情况7.2.3 字符的存储情况7.2.4 字符数组在内存中的存储情况7.3 向函数传递数组7.4 C++二维数组7.4.1 线与面——一维数组和二维数组7.4.2 二维数组的声明和初始化7.4.3 省略第一维的大小7.4.4 二维数组在内存中的存储情况7.4.5 向函数传递二维数组7.4.6 二维数组转化成一维数组7.5 方法指导7.6 习题第8章 内存里的快捷方式——指针8.1 什么是指针8.2 C++中指针变量的声明和使用8.2.1 指针的类型8.2.2 指针变量的声明8.2.3 获取地址和指针变量初始化8.2.4 特殊的值——NULL8.2.5 指针的使用——间接引用8.3 指针的操作8.3.1 指针的加减运算8.3.2 指针的关系运算8.4 指针与保护8.4.1 对内存只读的指针8.4.2 指针型常量8.5 指针与数组8.5.1 数组名的实质8.5.2 指针数组8.6 指针与函数8.6.1 指针作为参数8.6.2 指针作为返回值8.7 更灵活的存储——堆内存空间8.7.1 如何获得堆内存空间8.7.2 有借有还,再借不难——堆内存的回收8.8 方法指导8.9 习题第9章 自己设计的箱子——枚举和结构9.1 我的类型我做主——枚举类型9.2 设计一个收纳箱——定义结构类型9.3 C++结构与函数9.3.1 结构作为参数9.3.2 结构作为返回值9.4 C++结构数组与结构指针9.4.1 结构数组9.4.2 结构指针9.5 自行车的链条——链表9.6 C++链表的实现9.6.1 链表的创建和遍历9.6.2 链表的查询9.6.3 插入结点9.6.4 删除结点9.6.5 清除链表9.7 方法指导9.8 习题第二篇 实战程序设计第10章 高效阅读程序代码10.1 整体把握法10.1.1 阅读C++代码的顺序10.1.2 整体把握语意10.2 经验法10.3 模拟法10.4 方法指导10.5 习题第11章 调试程序代码技巧11.1 再谈变量11.1.1 标志符11.1.2 C++全局变量和局部变量11.1.3 静态局部变量11.1.4 变量的作用域11.1.5 变量的可见性11.2 C++头文件的奥秘11.2.1 如何创建一个头文件11.2.2 C++程序中头文件的作用11.2.3 头文件和源文件11.2.4 细说#include11.2.5 #include中尖括号和双引号的区别11.3 更快更好地完成程序调试11.3.1 如何检查语法错误11.3.2 常见语法错误及解决方法11.4 最麻烦的问题——运行时错误11.4.1 见识运行时错误11.4.2 查找错误点11.5 调试工具——Debug11.5.1 设置和移除断点11.5.2 Go语句11.5.3 Debug窗口11.5.4 Watch窗口11.5.5 用Debug找到错误11.6 方法指导11.7 习题第12章 编写程序技巧12.1 程序设计的基本步骤12.2 三类C++编程问题12.2.1 算法实现12.2.2 匹配实现12.2.3 功能实现12.3 函数的递归12.3.1 什么是栈12.3.2 函数

的调用机制12.3.3 小试牛刀——用递归模拟栈12.3.4 \*递归的精髓12.4 方法指导12.5 习题第三篇

面向对象的程序设计第13章 初识对象13.1 对象的定义13.2 一个字符串也是对象13.2.1 奇妙的点13.2.2 对字符串的操作13.3 面向对象特点一：封装性13.4 从数组到向量13.4.1 向量的性能13.4.2 万用的模板13.4.3 对向量的操作13.5 方法指导13.6 习题第14章 再识对象14.1 类是一种数据类型14.1.1 类与结构14.1.2 类的声明与定义14.2 公有和私有14.3 成员函数14.3.1 成员函数的声明14.3.2 常成员函数14.3.3 成员函数的重载14.3.4 成员函数的定义14.4 对象、引用和指针14.4.1 对象的引用14.4.2 对象指针14.5 方法指导14.6 习题第15章 造物者与毁灭者——对象生灭15.1 麻烦的初始化15.2 造物者——构造函数15.2.1 构造函数的声明与定义15.2.2 带参数的构造函数15.3 先有结点，还是先链表15.4 “克隆”技术——拷贝构造函数15.4.1 拷贝构造函数15.4.2 默认拷贝构造函数15.4.3 拷贝构造函数存在的意义15.5 毁灭者——析构函数15.6 方法指导15.7 习题第16章 共有财产·好朋友·操作符16.1 有多少个结点16.1.1 静态成员数据16.1.2 静态成员数据的初始化16.1.3 静态成员函数16.2 类的好朋友——友元16.2.1 友元类16.2.2 友元函数16.2.3 使用友元的利与弊16.3 多功能的操作符——操作符的重载16.3.1 操作符作为成员函数16.3.2 操作符作为友元函数16.3.3 又见加加和减减16.4 方法指导16.5 习题第17章 父与子——继承17.1 剑士·弓箭手·法师的困惑17.2 面向对象特点二：继承性17.3 继承的实现17.3.1 私有和保护17.3.2 一个简单的例子17.3.3 继承的方式17.4 子类对象的生灭17.4.1 子类对象的构造17.4.2 子类对象的析构17.5 继承与对象指针17.5.1 父类指针与子类对象17.5.2 猜猜它是谁——覆盖17.6 面向对象特点三：多态性17.7 多态与虚函数17.7.1 多态的实现17.7.2 无法实现多态的虚函数17.8 虚函数与虚析构函数17.9 抽象类与纯虚函数17.10 多重继承17.11 方法指导17.12 习题第18章 再谈输入与输出18.1 cout和cin真正含义18.2 输入输出的重定向18.2.1 输入重定向18.2.2 输出重定向18.2.3 无法被重定向的cerr18.3 文件的输入与输出18.4 更巧妙地输入和输出18.4.1 能整行输入的getline18.4.2 能读取判断末尾的eof18.4.3 能计数的gcount18.4.4 能设置域宽的width18.5 插入操作符的重载18.5.1 插入操作符18.5.2 插入操作符的常用重载方式18.6 方法指导18.7 习题第19章 万用的模板19.1 函数模板19.1.1 声明与定义函数模板19.1.2 函数模板与重载19.2 类模板19.2.1 类模板的声明和定义19.2.2 链表类模板实例19.3 方法技巧19.4 习题第20章 异常的处理20.1 亡羊也要补牢——程序出错处理20.2 处理异常20.2.1 尽力尝试——try语句20.2.2 抓住异常——catch语句20.3 抛出异常——throw语句20.4 方法指导20.5 习题附录A 常用保留字列表附录B 常见编译错误和解决方法附录C 参考答案第2章第3章第4章第5章第6章第7章第8章第9章第10章第11章第12章第13章第14章第15章第16章第17章第18章第19章第20章附录D 参考文献

## 章节摘录

第一篇 过程化的程序设计 第1章 良好的学习开端 本章主要讲述学习程序设计前需要了解的一些知识和学习程序设计的方法，并且对c++作简要的介绍。

读者学好这一章，对日后的学习能够起到事半功倍的效果。

1.1 软件与程序 随着计算机的普及和科学技术的发展，无纸化办公、计算机辅助设计（CAD，CompeerAidedDesign）和计算机辅助制造（CAM，CompeerAidedManufacture）已经渐渐走进我们的日常生活中。

有了计算机的帮助，我们的工作效率得到明显的提升，设计人员只需要把数据输入计算机，就能显示出精确的结果，例如一个三维立体模型。

当我们使用计算机的时候，有没有想过人类是如何让计算机做这些工作的呢？

其实，我们平时对计算机进行的操作是在与计算机软件（Software）打交道。

计算机之所以能够帮助人类工作，离不开软件的支持。

那么软件到底是什么？

其实它是看不见摸不着，但却能够通过计算机为用户所用的一种东西。

打一个比方，计算机的各种硬件设备（Hardware）就像是人的肌肉，而软件就像是人的灵魂。

少了软件这个灵魂，那么计算机只是一堆废铜烂铁。

人们通过编写一款软件，来让计算机做一些事情。

像我们用的Windows、Word、QQ等都是软件。

那么，软件和我们所说的程序（Program）又有着什么样的关系呢？

首先，要弄清什么是程序。

从初学者比较容易理解的角度说，程序是计算机执行一系列有序的动作的指令集合。

通过一个程序，可以使计算机完成某一类有着共同特点的工作，如求解一个一元二次方程，或是找出一组数里面最大的一个数。

而一款软件，往往是由若干个相关的程序、运行这些程序所需要的数据和相关文档（如帮助文档）等多个文件组成的。

因此，要设计出一款软件，就必须从程序设计开始。

1.2 程序设计要做什么 很多初学者会不解：程序设计到底是要做什么呢？

我们该如何让计算机帮助解决问题呢？

其实，要解决一些看似不同的问题，可以归纳为一种确定的过程和方法。

我们把这种能够在有限的步骤内解决一类问题的过程和方法称为算法（Algorithm）。

下面，我们以解一元二次方程为例，介绍求解的算法。

## <<易学C++>>

### 编辑推荐

《易学C++》的内容涵盖了绝大部分常用的C++知识，可以作为大学计算机专业或非计算机专业的程序设计入门教材，也可供计算机爱好者自学使用。

#### 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>