

<<好学的C++程序设计>>

图书基本信息

书名：<<好学的C++程序设计>>

13位ISBN编号：9787115283092

10位ISBN编号：7115283095

出版时间：2012-8

出版单位：人民邮电出版社

作者：张祖浩，沈天晴 编著

页数：318

字数：513000

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>

<<好学的C++程序设计>>

内容概要

张祖浩等编著的《好学的C++程序设计》内容与大学课程“C++程序设计”基本相同，内容包括基本数据和别名指针、运算、流程控制、函数、数组、枚举类型与结构类型、类和对象、继承与派生、多态性、输入/输出流和上机操作等。

“C++程序设计”是高校普遍开设的一门计算机核心基础课程，同时也是一门非常难懂的课。为此，作者采取了多方面的革新措施，作出独到阐述，达到增强系统性、提高可读性的目的，驱除团团迷雾，化解层层难点。

使其既进得了中学生书房，使中学生能先修登上“C++殿堂”；又上得了大学生课堂，使大学生对难点迷雾豁然开朗。

《好学的C++程序设计》可用作中学生先修或选修课程“C++程序设计”的教学用书，可用作中学生兴趣小组的学习用书，也可用作中学生自学

“C++程序设计”用书或课余读物。

《好学的C++程序设计》还可用作高等院校本、专科相关专业学习“C++程序设计”课程的教学用书，或编程爱好者自学用书及教学参考书。

<<好学的C++程序设计>>

书籍目录

第1章 概论

1.1 计算机怎样计数

1.1.1 乒乓球场上的计分牌

1.1.2 二进制怎样表示全正数和正负数

1.1.3 计算机中二进制数怎样存储

1.2 计算机怎样对数据进行处理

1.2.1 把算法形成的程序输入计算机

1.2.2 用什么语言输入计算机能懂

1.3 C++程序怎样写出和运行

1.3.1 举个C++程序简例看看

1.3.2 C++程序从写出到运行的几个步骤

1.3.3 本书前几章内容的安排

1.4 习题

第2章 对基本数据分类存储和访问

2.1 数据分类入驻存储空间听候处理

2.1.1 基本数据分哪些类

2.1.2 什么是数据变量的存储空间、长度和取值范围

2.1.3 数据常量怎样表示

2.1.4 数据变量首次出场必须声明属何类型

2.1.5 怎样对数据变量进行访问

2.2 别名竟然登上大雅之堂

2.2.1 别名变量概念

2.2.2 别名变量怎样声明

2.2.3 用别名对变量进行访问例

2.3 循址访问是怎样的

2.3.1 “牧童遥指杏花村”的启发

2.3.2 指针变量概念

2.3.3 指针变量怎样声明和赋值

2.3.4 用指针所指对所指变量进行访问

2.3.5 基本类型变量的指针

2.4 对数据变量怎样保护

2.4.1 可用const声明常值数据变量

2.4.2 可在声明中用const对指针和别名进行限定

2.5 习题

第3章 数据的运算及简单输入/输出运算

3.1 运算符和表达式

3.1.1 运算符

3.1.2 表达式

3.2 算术运算符和表达式

3.2.1 基本算术运算符

3.2.2 除表达式a/b

3.2.3 取余表达式a%b

3.3 自增自减运算符和表达式

3.3.1 自增自减运算符

3.3.2 自增自减表达式

<<好学的C++程序设计>>

- 3.4 关系运算符和表达式
 - 3.4.1 关系运算符
 - 3.4.2 关系表达式
 - 3.5 逻辑运算符和表达式
 - 3.5.1 逻辑运算符
 - 3.5.2 逻辑表达式
 - 3.6 位运算符和表达式
 - 3.6.1 位运算符
 - 3.6.2 a|b是按位“或”表达式
 - 3.6.3 其余的位运算表达式
 - 3.7 条件运算符和表达式
 - 3.7.1 条件运算符
 - 3.7.2 条件表达式
 - 3.8 赋值运算符和表达式
 - 3.8.1 赋值运算符
 - 3.8.2 赋值表达式
 - 3.9 逗号运算符和表达式
 - 3.9.1 逗号运算符
 - 3.9.2 逗号表达式
 - 3.10 基本数据混合运算时类型的转换
 - 3.10.1 隐性类型转换
 - 3.10.2 强迫类型转换
 - 3.11 指针的增减运算
 - 3.11.1 指针的增减是什么意思
 - 3.11.2 指针的整数增减可比作走步
 - 3.11.3 小小一例竟使多方受益
 - 3.11.4 指针变量自增减也可比作走步
 - 3.11.5 指针两种走步的比较
 - 3.12 简单的输入/输出运算
 - 3.12.1 输入流和输出流
 - 3.12.2 提取运算符“>>”和插入运算符“<<”
 - 3.12.3 提取表达式和插入表达式
 - 3.12.4 基本数据的输入/输出
 - 3.13 对输入/输出默认格式不如意怎么办
 - 3.13.1 可用单项格式控制符实现如意的单项格式
 - 3.13.2 用单项格式控制符实现单项格式例
 - 3.14 习题
- 第4章 程序流程怎样控制
- 4.1 程序流程基本结构有哪几种
 - 4.1.1 三种基本结构
 - 4.1.2 两种特殊语句
 - 4.2 选择结构是怎样的
 - 4.2.1 if语句(又叫条件语句)
 - 4.2.2 switch语句(开关语句)
 - 4.3 循环结构是怎样的
 - 4.3.1 盲童数苹果的故事
 - 4.3.2 循环基本概念

<<好学的C++程序设计>>

- 4.3.3 while语句
- 4.3.4 do while语句
- 4.3.5 for语句
- 4.3.6 循环结构的嵌套——以鸡鸭天天下蛋为例
- 4.3.7 三种循环语句的比较
- 4.4 其他控制语句和函数
 - 4.4.1 跳出状态的break语句
 - 4.4.2 只中断本次循环的continue语句
 - 4.4.3 收尾并终止程序的exit函数
- 4.5 习题
- 第5章 分担任务的得力助手——函数
 - 5.1 函数是怎么回事
 - 5.1.1 “自顶向下，逐步细化，函数分担”的模块化程序设计
 - 5.1.2 函数怎样定义
 - 5.2 怎样调用函数执行任务
 - 5.2.1 函数原型声明
 - 5.2.2 函数的调用
 - 5.2.3 函数自己调用自己——递归调用
 - 5.3 函数内外变量的作用域和生存期
 - 5.3.1 作用域就是变量起作用的范围
 - 5.3.2 变量的死活要看生存期
 - 5.4 函数内外数据怎样传递
 - 5.4.1 可通过函数参数进行数据传递
 - 5.4.2 可用具有默认值的参数传递
 - 5.4.3 可通过函数返回值进行数据传递
 - 5.4.4 可通过全局变量进行数据传递
 - 5.5 什么是内联函数、重载函数、函数模板和模板函数
 - 5.5.1 内联函数
 - 5.5.2 重载函数
 - 5.5.3 函数模板和模板函数
 - 5.6 指针也能对函数进行调用
 - 5.6.1 指向函数的指针
 - 5.6.2 函数指针变量的声明、初始化和赋值
 - 5.6.3 可用函数指针调用函数
 - 5.7 函数的多文件组织
 - 5.7.1 模块化程序设计
 - 5.7.2 源文件之间的访问
 - 5.7.3 头文件
 - 5.7.4 多文件组织怎样编译和链接
 - 5.8 编译的预处理
 - 5.8.1 #include指令
 - 5.8.2 #define和#undef指令
 - 5.9 条件编译
 - 5.9.1 条件编译有三种形式
 - 5.9.2 关于条件编译的说明
 - 5.10 习题
- 第6章 同类型数据排成队——数组

<<好学的C++程序设计>>

6.1 一维数组

6.1.1 一维数组怎样声明和初始化

6.1.2 用下标表示形式对一维数组元素进行访问

6.1.3 可用数组名表示一维数组元素

6.1.4 可用一维数组名调用函数处理一维数组

6.2 二维数组

6.2.1 二维数组怎样声明和初始化

6.2.2 用下标表示形式对二维数组元素进行访问

6.2.3 可用数组名表示二维数组元素

6.2.4 可用二维数组名调用函数处理二维数组

6.3 对字符串怎样处理

6.3.1 可用一维字符数组处理字符串

6.3.2 可用库函数处理字符串

6.4 对字符串组怎样处理

6.4.1 可用一维字符指针数组处理字符串组

6.4.2 可用一维字符指针数组名调用函数处理字符串组

6.5 合适的存储区想要就能有吗

6.5.1 可用new运算符申请动态配给存储区

6.5.2 可用delete运算符撤放动态配给的存储区

6.5.3 可给临时输入的姓名字符串动态配给“经济适用房”

6.6 应用范例——建立姓名录排序和输出

6.6.1 问题的提出

6.6.2 分析

6.6.3 源程序及说明

6.7 习题

第7章 枚举类型与结构类型

7.1 枚举类型是怎样的

7.1.1 枚举类型的定义

7.1.2 枚举型变量的声明、初始化和赋值

7.1.3 可调用函数对枚举型变量输入和输出

7.1.4 “酒楼点菜”一枚举元素的组合状态

7.2 应用范例——C++用格式状态字表明输入/输出格式的组合状态

7.2.1 将各种输入/输出格式定义为各个枚举元素

7.2.2 用格式状态字flag表明多项格式的组合状态

7.2.3 用格式状态字调用组合格式控制符实现多项格式的组合采用

7.3 结构类型是怎样的

7.3.1 结构类型的定义

7.3.2 结构型变量的声明、初始化和赋值

7.3.3 可对结构型变量的成员进行访问

7.3.4 可用结构型数组处理结构型变量

7.4 什么是链表

7.4.1 链表怎样组成

7.4.2 怎样对链表结点进行访问

7.4.3 对链表进行操作的必做预习题

7.5 可调用函数对链表进行各项操作

7.5.1 调用函数把一个结点插入顺序链表

7.5.2 调用函数建立一条有序新链表

<<好学的C++程序设计>>

- 7.5.3 调用函数输出链表各结点数据
- 7.5.4 调用函数删除链表上具有指定值的一个结点
- 7.5.5 调用函数撤放链表全部结点动态配给的存储空间
- 7.6 应用范例——调用函数建立有序链表和删除指定结点
 - 7.6.1 问题的提出
 - 7.6.2 分析
 - 7.6.3 源程序及说明
- 7.7 用typedef可定义某类型的又一个标识符
 - 7.7.1 用typedef定义某类型又一个标识符例
 - 7.7.2 用typedef定义某类型又一个标识符的方法步骤
- 7.8 习题
- 第8章 类和对象
 - 8.1 从面向过程到面向对象
 - 8.2 类具体是怎样的
 - 8.2.1 类的定义
 - 8.2.2 对类内各成员访问的控制规则
 - 8.2.3 类的引用性声明
 - 8.2.4 面向对象程序设计的多文件组织
 - 8.3 对象具体是怎样的
 - 8.3.1 对象的声明
 - 8.3.2 同类对象之间可以整体赋值
 - 8.3.3 对对象成员的访问
 - 8.3.4 地下工作者——本类指针this
 - 8.3.5 对对象数据成员置值的两种方法
 - 8.4 迎接对象诞生的函数——构造函数
 - 8.4.1 构造函数的作用特点和定义形式
 - 8.4.2 构造函数的重载
 - 8.4.3 构造函数和new运算符
 - 8.5 送别对象撤销的函数——析构函数
 - 8.5.1 析构函数作用特点和定义形式
 - 8.5.2 一定要定义析构函数吗
 - 8.6 非要定义构造函数和复制构造函数吗
 - 8.6.1 默认的构造函数和复制构造函数
 - 8.6.2 浅复制和深复制
 - 8.7 类属成员——类的静态成员
 - 8.7.1 静态数据成员
 - 8.7.2 静态成员函数
 - 8.8 类可结交“亲密好友”
 - 8.8.1 友元的概念
 - 8.8.2 运用友元的一个例题
 - 8.8.3 友元声明的一般形式
 - 8.8.4 关于友元的几点说明
 - 8.9 什么叫类嵌套
 - 8.9.1 类嵌套关系
 - 8.9.2 嵌套类实例
 - 8.9.3 嵌套类构造函数定义的一般形式
 - 8.10 类模板和模板类

<<好学的C++程序设计>>

- 8.10.1 什么叫类模板和模板类
- 8.10.2 类模板怎样定义
- 8.10.3 类模板怎样使用
- 8.11 应用范例——面向对象构建学生成绩链表
 - 8.11.1 问题的提出
 - 8.11.2 类设计
 - 8.11.3 创建链表的思路
 - 8.11.4 源程序及说明
- 8.12 习题
- 第9章 继承与派生
 - 9.1 继承与派生是怎么回事
 - 9.1.1 继承、派生、基类、派生类概念
 - 9.1.2 基类和派生类的构成形式
 - 9.1.3 派生类成员的组成和身份的确定
 - 9.2 派生类
 - 9.2.1 举个派生类简例
 - 9.2.2 派生类的定义形式
 - 9.2.3 派生类成员存储空间、身份及访问
 - 9.2.4 派生类的构造函数和析构函数
 - 9.3 对派生类中同名成员二义性的处理
 - 9.3.1 类名加域运算符::处理法
 - 9.3.2 同名覆盖原理
 - 9.3.3 对共同基类多级多脉继承中同名成员的处理
 - 9.3.4 用虚基类避免一个数据多种版本
 - 9.4 类的赋值兼容
 - 9.4.1 公有派生类对象可以顶替基类对象
 - 9.4.2 公有派生类对基类的赋值兼容
 - 9.4.3 公有派生类对象怎样顶替基类对象
 - 9.4.4 类的赋值兼容规则的实质
 - 9.5 应用范例——半工半读学生信息管理系统
 - 9.5.1 问题的提出
 - 9.5.2 类设计
 - 9.5.3 源程序及说明
 - 9.5.4 源程序呼喊改进
 - 9.5.5 虚函数的提出
 - 9.6 习题
- 第10章 多态性
 - 10.1 什么叫做静态联编和动态联编
 - 10.2 虚函数是怎么回事
 - 10.2.1 虚函数的定义
 - 10.2.2 直接访问同名成员函数时的接访规则
 - 10.2.3 虚函数在实现动态联编多态性中的运用例
 - 10.2.4 虚析构函数
 - 10.3 什么是纯虚函数和抽象类
 - 10.3.1 纯虚函数
 - 10.3.2 抽象类
 - 10.3.3 抽象基类例

<<好学的C++程序设计>>

- 10.4 运算符重载是怎样的
 - 10.4.1 运算符重载概念
 - 10.4.2 运算符重载要求
 - 10.4.3 运算符重载函数
- 10.5 怎样用成员函数实现运算符重载
 - 10.5.1 成员函数实现双目运算符重载
 - 10.5.2 成员函数实现单目运算符重载
- 10.6 怎样用友元函数实现运算符重载
 - 10.6.1 友元函数实现双目运算符重载
 - 10.6.2 友元函数实现单目运算符重载
- 10.7 对象运算中怎样进行类型转换
 - 10.7.1 转换构造函数
 - 10.7.2 类型转换函数
 - 10.7.3 对象运算中类型转换例
- 10.8 应用范例——类型转换应用于时间运算中
 - 10.8.1 问题的提出
 - 10.8.2 设计思路
 - 10.8.3 源程序及说明
- 10.9 习题
- 第11章 输入/输出流类体系
 - 11.1 什么是流类和流
 - 11.1.1 流类概念(端口、模式和成员函数)
 - 11.1.2 缓冲流
 - 11.1.3 流类体系
 - 11.2 基本流类体系是怎样的
 - 11.2.1 基本流类体系组成
 - 11.2.2 基本流类体系各组成部分简介
 - 11.3 什么是I/O标准流
 - 11.3.1 I/O标准流概念
 - 11.3.2 I/O标准流的端口和模式
 - 11.3.3 常用于输入的成员函数
 - 11.3.4 常用于输出的成员函数
 - 11.3.5 用于格式控制的成员函数
 - 11.3.6 用于检验出错的成员函数
 - 11.4 文件流类体系是怎样的
 - 11.4.1 文件流类体系组成
 - 11.4.2 文件流类体系各组成部分简介
 - 11.5 什么是I/O文件流
 - 11.5.1 I/O文件流概念
 - 11.5.2 I/O文件流的建立, 端口和模式的确定
 - 11.5.3 用于建立和关闭I/O文件流的成员函数
 - 11.5.4 I/O文本文件流常用的成员函数
 - 11.5.5 I/O二进制文件流常用的成员函数
 - 11.6 应用范例——文件中建立平方根表
 - 11.6.1 问题的提出
 - 11.6.2 分析
 - 11.6.3 源程序及说明

<<好学的C++程序设计>>

11.7 习题

第12章 简述VC++6.0编程环境的上机操作

12.1 编制实现单文件应用程序的方法

12.1.1 编制一个简单程序

12.1.2 为简单程序修改错误

12.2 程序设计中的多文件组织

12.3 怎样查找程序运行中的错误

12.3.1 查找程序运行的错误点

12.3.2 VC6.0有哪些调试工具

12.3.3 神奇的单步调试

12.4 实验题

附录A ASCII码表

附录B 习题参考答案

<<好学的C++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>