

<<Visual C++程序设计教程>>

图书基本信息

书名：<<Visual C++程序设计教程>>

13位ISBN编号：9787305056994

10位ISBN编号：7305056995

出版时间：2009-1

出版时间：南京大学出版社

作者：赵璐等著

页数：313

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>

## <<Visual C++程序设计教程>>

### 前言

由Dennis Ritchie于20世纪70年代创建的c语言以简洁、高效和良好的移植性将程序设计带入了一个新的时代，而Bjarne Stroustrup在C语言的基础之上设计的C++上语言，通过向C语言加入面向对象的特性为程序设计开创了一个新纪元。

作为C语言的超集，c++为程序设计员提供了强大的功能，可以产生模块化程度高、重用性和可维护性好的程序。

目前c++在各个领域都得到了广泛的应用，成为最流行的程序设计语言之一。

本书面向程序设计初学者，以Visual C++6.0为编程环境介绍了C++程序设计的基础知识。

全书共15章，分两部分：第一部分为VC++程序设计基础，共8章，循序渐进地介绍了VC++的基本语法及结构化程序设计的基本知识和技巧；第二部分为面向对象的程序设计，共7章，介绍了面向对象的基本概念和OOP程序设计的基本知识。

另外，本书在附录中收录了ASCII码表、C++关键字一览表、基本运算符优先级和结合性一览表以及常用函数简表，便于读者进行相关方面的查阅和参考。

作为程序设计的入门教材，编者在编写过程中力求从读者的角度出发，由浅入深地安排内容、简洁而准确地阐述概念，同时配以详实的图表。

在例题的组织上，除了紧密围绕知识点、由简到繁地引入例题，更是对例题进行了简单的分析或点评，突出例题中的知识点和编程技巧，便于读者理解和学习。

在每章内容后，还安排了大量的习题，这些习题从易到难地帮助读者在理解、掌握基本概念和知识点的基础上，一步一步提高编程能力。

## <<Visual C++程序设计教程>>

### 内容概要

《Visual C++程序设计教程》面向程序设计初学者，以Visual C++6.0为编程环境介绍了C++程序设计的基础知识。

全书共15章，分两部分：第一部分为VC++程序设计基础，共8章，循序渐进地介绍了VC++的基本语法及结构化程序设计的基本知识和技巧；第二部分为面向对象的程序设计，共7章，介绍了面向对象的基本概念和OOP程序设计的基本知识。

另外，《Visual C++程序设计教程》在附录中收录了ASCII码表、C++关键字一览表、基本运算符优先级和结合性一览表以及常用函数简表，便于读者进行相关方面的查阅和参考。

《Visual C++程序设计教程》可作为普通高校程序设计课程的教材，也可作为程序设计初学者的自学用书。

## &lt;&lt;Visual C++程序设计教程&gt;&gt;

## 书籍目录

1 C++概述1.1 C++简介1.2 C++程序示例1.3 C++程序的实现2 C++语言基础2.1 标识符2.2 数据类型2.3 变量2.4 常量2.5 运算符和表达式3 C++程序结构3.1 程序设计和算法3.2 C++程序的语句3.3 顺序结构3.4 选择结构3.5 循环结构3.6 控制语句3.7 程序举例4 函数4.1 概述4.2 定义函数的一般形式4.3 函数的调用4.4 函数参数和函数的返回值4.5 函数的原型声明4.6 内联函数4.7 函数的嵌套调用和递归调用4.8 函数的重载4.9 变量的作用域4.10 变量的存储类型4.11 编译预处理5 数组5.1 概述5.2 一维数组5.3 二维数组5.4 数组作为函数的参数5.5 字符数组6 结构体、共同体和枚举型6.1 结构体类型6.2 共同体6.3 枚举类型6.4 用typedef定义类型7 指针7.1 指针与地址7.2 指针变量7.3 指针与数组7.4 指针与字符串7.5 指针与函数7.6 指向指针的指针变量7.7 指针数据类型小结7.8 动态分配和撤消内存的运算符new和delete7.9 引用8 链表8.1 概述8.2 创建结点8.3 建立链表8.4 对链表的操作8.5 链表的程序举例9 类和对象9.1 面向对象的程序设计方法简介9.2 类和对象的基本概念9.3 类的成员函数9.4 this指针10 构造函数与析构函数10.1 构造函数10.2 析构函数10.3 构造函数和析构函数的调用时机10.4 对象成员11 友元与静态成员11.1 友元11.2 静态成员12 运算符重载12.1 运算符重载的引入12.2 运算符重载的基本概念12.3 通过成员函数实现运算符重载12.4 通过友元函数实现运算符重载12.5 “++”自增运算符的重载12.6 “=”赋值运算符的重载13 继承和派生类13.1 继承13.2 访问控制13.3 初始化基类成员13.4 冲突、支配规则和赋值兼容规则13.5 虚基类14 多态性与虚函数14.1 静态联编与动态联编14.2 虚函数14.3 纯虚函数和抽象类14.4 抽象类的实例15 C++流和文件流15.1 C++流的概念15.2 格式化I/O15.3 重载I/O运算符15.4 文件流附录一 ASCII码表完整版(十进制)附录二 C++关键字一览表附录三 运算符优先级、结合性一览表附录四 常用函数和头文件一览表参考书目

## 章节摘录

**指针** 指针是Vc++中广泛使用的一种数据类型，利用指针可以方便地操作数组和字符串，在调用函数时能得到更多的值，并能像汇编语言一样处理内存地址，从而编出精练而高效的程序。

**7.1指针与地址** 程序和数据在内存中都是由CPU来执行和处理的，内存就好比是一座旅馆，里面有许许多多的小房间，每个房间又按顺序编了号。

计算机中要处理的数据都是存放在内存的房间里，好比是住在房间中的旅客。

只要给出了房间的编号就能找到所要访问或修改的数据。房间的编号就称为内存的地址。

内存中“房间”的大小是规定的，每间“房间”只能存储一个字节（Byte）的数据。

比如一个字符型变量，就可以将其值存储在一个“房间”里。

但有时候，某些类型的变量需要比较大的空间。

比如一个双精度型的实数，一个“房间”是放不下的，而是需要8间“房间”的空间才能放得下。

于是计算机就把8间“房间”拼起来，每间“房子”存入这个实数的一部分数据。

而这连接起来的8间“房间”构成了一个存放双精度型实数的变量。

如果在程序中定义了一个变量，系统在编译时就会根据变量的类型为其分配一定的内存单元，并将这些内存单元首字节的编号称为该变量的地址。

在程序中一般是通过变量名来对内存单元进行存取操作的，因此变量名实际上就是给内存单元取的一个容易记忆的名字，访问变量时首先应根据变量名与内存单元之间的对应关系找到其内存地址，然后对其中的数据进行访问。

设有定义“`Int a;`”，编译时系统分配地址为10000、10001、10002、10003的4个字节给整型变量a。

如果有语句“`a = 10;`”，在执行时，根据变量名与地址的对应关系，找到变量a的地址10000，将数值10保存在从10000开始的4个字节的存储单元中。

这种按变量名存取变量值的方式称为直接存取方式，或直接访问方式。

此外，在VC++中还可以采用另一种称为间接访问的方式。

假设定义了一个变量`a_pointer`，专门用来存放一个整型变量的地址，那么可以通过下面的语句将a的起始地址（10000）存放到`a_pointer`中。

`a_pointer = &a;` “&”是取地址运算符，`&a`表示变量a的地址。

执行此语句后，`a_pointer`的值就是10000（即变量a所占用存储单元的起始地址）。

若要取变量a的值，可以先找到存放“a的地址”的变量`a_pointer`，从中取出a的地址（即10000），然后到10000开始的4个字节中取出a的值，这种访问方式就称为间接访问方式。

<<Visual C++程序设计教程>>

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>