

<<C++程序设计>>

图书基本信息

书名：<<C++程序设计>>

13位ISBN编号：9787302182979

10位ISBN编号：7302182973

出版时间：2008-9

出版时间：清华大学出版社

作者：赵清杰

页数：229

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++程序设计>>

内容概要

本书系统地讲解了C++语言的基本语法及编程方法，介绍C++标准库特别是STL的主要组件及应用；内容包括C++语言概述及编程基础、函数与函数模板、类与类模板、运算符重载、继承与派生、多态、异常处理以及C++标准库等。

针对初学者的特点，书中主要结合实例讲解基本概念和编程方法，力求通过简洁的实例让读者快速掌握C++语言，并能够熟练使用C++标准库进行程序设计。

本书内容全面、语言简洁、重点突出、实用性强，既适合于作为高等学校的教材，也适合作为培训班教材及自学参考书。

书籍目录

第1章 C++语言概述 1.1 C++语言的特点 1.2 C++标准库简介 1.3 简单的C++程序 1.4 程序的编辑、编译、连接与运行 1.5 小结 习题一第2章 C++编程基础 2.1 C++的词法规则 2.1.1 字符集 2.1.2 词汇 2.2 C++的数据类型 2.2.1 基本类型 2.2.2 常量与变量 2.2.3 自定义数据类型 2.2.4 扩展数据类型 2.2.5 类型转换 2.2.6 typedef与typeid 2.3 表达式与语句 2.3.1 运算符与表达式 2.3.2 语句 2.4 预处理命令 2.4.1 宏定义命令 2.4.2 文件包含命令 2.4.3 条件编译命令 2.5 名字空间 2.5.1 名字空间声明 2.5.2 使用名字空间 2.5.3 标准名字空间std 2.6 小结 习题二第3章 函数与函数模板 3.1 函数的定义与声明 3.2 函数调用 3.2.1 如何调用函数 3.2.2 参数传递 3.2.3 函数的返回类型 3.2.4 嵌套调用与递归调用 3.2.5 如何调用库函数 3.3 函数指针 3.4 static函数 3.5 inline函数 3.6 函数重载 3.7 带默认形参值的函数 3.8 函数模板 3.8.1 函数模板的定义与使用 3.8.2 函数模板重载 3.8.3 函数模板专门化 3.8.4 使用标准库中的函数模板 3.9 小结 习题三第4章 类与类模板 4.1 类与类对象的定义 4.1.1 类的定义 4.1.2 类对象： 4.1.3 类的封装性和信息隐藏 4.2 构造函数与析构函数 4.2.1 构造函数 4.2.2 析构函数 4.2.3 构造与析构的顺序 4.3 赋值成员函数 4.4 静态成员 4.4.1 静态数据成员 4.4.2 静态成员函数 4.5 常成员 4.5.1 常数据成员 4.5.2 常成员函数 4.5.3 mutable 4.6 指向成员的指针 4.6.1 成员指针的定义与使用 4.6.2 如何得到成员函数的地址 4.7 组合类 4.8 友元 4.8.1 友元函数 4.8.2 友元类 4.9 类模板 4.9.1 类模板的定义与使用 4.9.2 类模板专门化 4.9.3 作为函数的参数及返回类型 4.9.4 使用标准库中的类模板 4.10 小结 习题四第5章 运算符重载 5.1 运算符重载的概念 5.2 运算符重载的规则 5.3 运算符重载的两种形式 5.3.1 重载为类的成员函数 5.3.2 重载为类的友元函数 5.3.3 两种重载方式讨论 5.4 特殊运算符重载举例 5.4.1 类型转换运算符 5.4.2 复合赋值运算符 5.4.3 自增和自减运算符 5.4.4 流提取运算符和流插入运算符 5.5 函数对象 5.6 小结 习题五第6章 继承与派生 6.1 基类与派生类 6.2 对基类成员的访问控制 6.2.1 公有继承 6.2.2 私有继承 6.2.3 保护继承 6.3 派生类的构造函数与析构函数 6.3.1 构造函数 6.3.2 析构函数 6.4 组合与继承的选择 6.5 多继承中的歧义 6.6 虚基类 6.7 类模板的继承与派生 6.8 小结 习题六第7章 多态 7.1 多态性概述 7.2 虚函数 7.2.1 虚函数的声明与应用 7.2.2 虚析构函数 7.3 C++如何实现动态绑定 7.4 纯虚函数与抽象类 7.5 小结 习题七第8章 异常处理 8.1 异常处理概述 8.2 异常处理的实现 8.3 带异常声明的函数 8.4 成员函数抛出异常 8.4.1 一般成员函数抛出异常 8.4.2 构造函数抛出异常 8.4.3 析构函数抛出异常 8.5 标准库中的异常类型 8.6 小结 习题八第9章 C++标准库 9.1 标准库组织 9.2 容器 9.2.1 容器的成员 9.2.2 顺序容器 9.2.3 顺序容器适配器 9.2.4 关联容器 9.2.5 近容器 9.3 string 9.4 泛型算法 9.5 迭代器 9.5.1 迭代器的分类 9.5.2 使用迭代器 9.6 函数对象 9.7 流类 9.7.1 标准流 9.7.2 文件流 9.7.3 串流 9.7.4 重载提取运算符和插入运算符 9.7.5 输入/输出成员函数 9.7.6 输入/输出格式控制 9.8 数值计算 9.8.1 数学函数 9.8.2 向量计算 9.8.3 复数计算 9.8.4 泛型数值算法 9.8.5 随机数产生 9.9 小结附录A ASCII码表参考文献

<<C++程序设计>>

章节摘录

第1章 C++语言概述 本章简单介绍c++语言的特点以及C++标准库的主要构成。首先通过一个简单例子让初学者建立对C++程序的初步认识；然后介绍C++程序的编辑、编译、连接与运行方法，以及在Visual C++6.0开发环境下建立标准C++控制台应用程序的步骤。

本章涉及不少概念或名词，读者没有完全理解也无关系要，通过后面的学习会逐渐理解的。

1.1 C++语言的特点 20世纪80年代，AT&T贝尔实验室的Bjarne Stroustrup博士研发出了C++语言，他的出发点是为了让编程人员能够更容易、更快捷地编写出高质量的程序。后来，c++语言得到了广泛应用。

经过多年的不断完善和发展，1998年国际标准化组织公布了C++语言的国际标准，2002年又进行了修订，形成了今天的国际标准c++语言。

国际标准（包括语法和程序库）的建立，简化了C++语言的使用以及在不同平台之间的移植工作，使用户可以根据爱好选用不同的C++编译器。

C++语言是支持多种编程思想的程序设计语言。

它不仅支持过程式程序设计（procedural programming），还支持基于对象的设计（object—based programming）、面向对象的程序设计（object—oriented programming）以及泛型程序设计（generic programming）。

过程式程序设计通过一组算法建立问题的模型，程序的构成就是“程序—算法+数据”，这里的算法（或者操作 / 方法 / 过程）通过函数实现。

C语言就是一门支持过程式程序设计的语言。

C++语言能够兼容C语言，这样就可以保证原来的C库函数可以继续使用。

C++语言在C语言的基础上做了很多改进。

例如对类型要求更加严格，输入输出更加方便，增加了新的运算符，允许函数重载（overloading）和运算符重载，增加了引用（reference）类型，提出内嵌（inline）概念，提供常类型关键字const等。

总之，在支持过程式程序设计方面，C++语言比C语言更安全，功能更强，使用更方便。

C++语言支持基于对象的设计。

<<C++程序设计>>

编辑推荐

《C++程序设计》作者结合实例讲解了C++语言的基本概念和编程方法，介绍C++标准库特别是STL的主要组件及应用，并力求通过简洁的实例让读者快速掌握C++语言，并能够熟练使用C++标准库进行程序设计。

全书共分9个章节，具体内容包括C++语言概述及编程基础、函数与函数模板、类与类模板、运算符重载等。

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>