第一图书网, tushu007.com

<<程序设计>>

图书基本信息

书名：<<程序设计>>

13位ISBN编号：9787111390398

10位ISBN编号：7111390393

出版时间：2012-7

出版时间：机械工业出版社

作者：王立柱

页数：268

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

内容概要

C++主要包括两部分内容：C和C++的公共子集，C++直接支持的高级技术部分。
C++包含四种程序设计风格：过程化程序设计、数据抽象、面向对象程序设计和泛型程序设计。
《高等院校计算机专业人才能力培养规划教材：程序设计·从过程化到面向对象》从公共子集到高级技术部分逐步讲授，内容涉及：机器语言简介，基本数据类型，运算符和表达式，语句组结构，指针和数组，函数，模块化程序设计，C风格串，结构、联合、枚举，二维数组和指针，顺序表，单向链表，从C到C++的基本内容，从顺序表到顺序表类，从C风格串到String类串，Date类，继承和动态绑定，函数模板和向量类模板，链表类模板和适配器，C++的I/O流库，C++综合设计实例，命名空间。

《高等院校计算机专业人才能力培养规划教材：程序设计·从过程化到面向对象》可以作为高等院校计算机及相关专业本科生以C++作为第一门程序设计课程的教材，也可以作为计算机编程爱好者的自学教材和参考书。

书籍目录

出版者的话
编委会
丛书序言
前言
教学建议
第1章 机器语言简介
1.1 计算机组成及工作过程
1.2 计算机硬件和软件
1.3 机器语言程序
1.4 汇编语言
1.5 深入探讨-存储和算法是一对矛盾体
习题
第2章 基本数据类型
2.1 变量与字面值常量
2.2 整型
2.3 字符型
2.4 实型
2.5 布尔型
2.6 typedef名字
2.7 基本数据类型的深入探讨
2.7.1 字面值常量的意义
2.7.2 左值和右值
习题
第3章 运算符和表达式
3.1 表达式
3.2 关系操作符
3.3 逻辑操作符
3.4 自增自减操作符
3.5 赋值和复合赋值操作符
3.6 条件操作符
3.7 逗号操作符
3.8 复合表达式27
3.9 内部类型转换
3.9.1 赋值兼容性
3.9.2 表达式计算中的类型转换过程
3.9.3 强制类型转换
习题
第4章 语句组结构
4.1 选择结构
4.1.1 if-else语句
4.1.2 switch语句
4.2 循环结构
4.2.1 for语句
4.2.2 while语句
4.2.3 do-while语句

4.2.4 循环嵌套
4.3 其他流程控制语句
4.3.1 break语句
4.3.2 continue语句
习题
第5章 指针和数组
5.1 指针
5.1.1 指针和间接引用
5.1.2 指针定义中“*”的位置
5.1.3 指针加减整数的基本运算
5.2 数组
5.2.1 数组变量和数组指针
5.2.2 变量和长度为1的数组
5.2.3 数组和指针的关系
5.2.4 动态数组
5.3 指针和数组的应用设计举例
5.3.1 数组元素求和
5.3.2 求数组最小元素
5.3.3 选择排序
5.4 指针和数组的深入讨论
5.4.1 指针、数组、表达式和左值
5.4.2 数组变量指针和数组指针
习题
第6章 函数
6.1 函数定义与调用
6.2 值传递和指针传递
6.3 返回值与指针传递
6.4 函数声明（原型）
6.5 函数应用设计举例
6.5.1 选择排序
6.5.2 起泡排序
6.5.3 数制转换
6.5.4 折半查找
6.5.5 划分数组元素
6.5.6 删除重复数据
6.5.7 筛法求素数
6.5.8 Josephus问题
6.6 函数调用与变量的存储类别
6.6.1 自动局部变量
6.6.2 静态局部变量
6.6.3 外部变量
6.6.4 寄存器变量
6.7 const限定修饰符
6.7.1 const型变量
6.7.2 间接const型指针
6.7.3 const型指针
6.7.4 全const型指针

6.8 函数指针
6.9 递归
6.1 0函数的深入讨论
6.1 0.1 作为函数退出点的return语句
6.1 0.2 自变量指针不能作为返回值
6.1 0.3 函数表达式和左值
6.1 0.4 移动下标与移动指针
6.1 0.5 数组和指针参数
习题
第7章 模块化程序设计
7.1 全局外部函数
7.2 静态外部函数
7.3 全局外部变量
7.4 静态外部变量
7.5 编译预处理
7.5.1 无参宏指令
7.5.2 带参宏指令
7.5.3 条件编译指令
7.5.4 文件包含指令
7.6 模块化应用设计举例-数组的
输入/输出、排序和查找
习题
第8章 C风格串
8.1 字符串常量和字符串赋值
8.2 字符串基本操作函数声明
8.3 字符串基本操作函数实现
8.4 字符串应用设计举例
8.4.1 判断回文
8.4.2 删除空格字符
8.5 字符串的深入讨论
8.5.1 特殊字符串
8.5.2 再论字符串常量
习题
第9章 结构、联合、枚举
9.1 结构
9.1.1 结构定义
9.1.2 结构变量
9.1.3 结构变量的初始化和赋值
9.1.4 结构数组
9.1.5 结构的嵌套
9.1.6 结构返回值和指针传递
9.2 联合
9.3 枚举
9.4 结构应用设计举例
9.4.1 模拟洗牌
9.4.2 Date结构
9.4.3 三天打鱼，两天晒网

9.5 结构的深入讨论
习题
第10章 二维数组和指针
10.1 二维数组与二维数组指针
10.2 二维数组与一维数组
10.3 指针数组与二级指针变量（指针的指针）
10.4 指针数组与二维数组
10.5 二级指针指向的动态数组空间
10.6 二维数组和指针的应用设计举例-马鞍点
习题
第11章 顺序表
11.1 顺序表的定义
11.2 顺序表基本运算的实现
11.3 删除顺序表重复数据
11.4 数据抽象和准封装
习题
第12章 单向链表
12.1 单向链表的定义
12.2 单向链表基本运算的实现
12.3 逆置
习题
第13章 从C到C++的基本内容
13.1 C语言本身的问题
13.2 内联函数
13.3 运算符重载
13.4 函数重载
13.5 引用型
13.5.1 引用型概念的由来
13.5.2 引用型的定义和应用
13.6 提取符和插入符重载
13.7 默认参数
习题
第14章 从顺序表到顺序表类
14.1 顺序表和顺序表类
14.2 顺序表类基本运算的实现
14.3 增加的成员函数
14.3.1 复制构造函数
14.3.2 复制赋值运算符重载
14.3.3 下标运算符重载
14.4 构造函数与初始化
14.5 类定义
习题
第15章 从C风格串到String类串
15.1 String类的声明
15.2 String类基本运算的实现
15.2.1 构造函数
15.2.2 成员赋值运算符

15.2.3 成员转换
15.2.4 串连接
15.2.5 关系运算
15.2.6 求子串
15.2.7 子串插入
15.2.8 子串删除
15.2.9 下标运算符
15.2.1 0字符查找
15.2.1 1输入/输出
15.3 模式匹配
15.4 深入讨论
15.4.1 转换赋值运算符函数的替代
15.4.2 成员函数“ 类串+C串 ”的替代
15.4.3 explicit修饰符
习题
第16章 Date类
16.1 Date类的声明
16.2 Data类基本运算的实现
16.3 深入讨论
16.3.1 转换赋值成员函数的替代
16.3.2 静态数据成员和静态函数成员
习题
第17章 继承和动态绑定
17.1 参数初始化表
17.2 继承和受保护成员
17.2.1 继承
17.2.2 受保护成员
17.3 多态性和虚函数
17.4 虚析构函数
17.5 纯虚函数和抽象类
习题
第18章 函数模板和向量类模板
18.1 函数模板
18.2 顺序表类模板
18.3 向量类模板Vector
18.3.1 Vector定义
18.3.2 通用算法和迭代器
18.3.3 Vector的插入和删除函数
18.3.4 求素数
18.4 函数对象
18.5 深入讨论-函数模板实例化中的问题
习题
第19章 链表类模板和适配器
19.1 链表类模板List
19.2 适配器
19.2.1 链栈
19.2.2 链队列

19.2.3 优先级链队列
习题
第20章 C++的I/O流库
20.1 类ofstream、ifstream和fstream
20.2 流模式标志和流成员函数
20.3 文件和流的关联与解除关联
20.4 无格式读写
20.5 格式化输入/输出
20.5.1 设置流的格式化标志
20.5.2 格式输出函数
20.5.3 操作算子
20.6 文件错误处理
习题
第21章 C++综合设计实例
21.1 中缀表达式求值
21.2 事件驱动模拟
习题
第22章 命名空间
22.1 命名空间的定义
22.2 usingnamespace语句
22.3 命名空间的成员
22.4 命名空间的别名
习题
附录A命名规则
附录B语言操作符的优先级与结合性
参考文献

章节摘录

版权页： 插图： 变量作用域是指变量名应该在程序的哪一部分可以直接访问，或者通俗地说，在程序中的哪一部分是可见的。
作用域的边界有三种：块、函数和文件。
变量生命周期是指变量空间从创建到撤销的这段时间。
一个变量如果不在其生命周期，显然不在其作用域；即使在其生命周期，未必一直是可见的。
变量根据其作用域和生命周期不同分为不同的存储类别：（自动）局部变量、静态局部变量、外部变量、寄存器变量、动态分配单元。
变量存储类别与变量类型不同：类型是一种存储模式，是按数据的存储空间大小、存储格式和基本操作来分类的，有整型、浮点型、字符型等。
同一类型的变量可以是不同存储类别，不同类型的变量可以同属一个存储类别。
6.6.1 自动局部变量 函数的形参和在函数体内定义的变量统称为局部变量。
局部变量又分为自动局部变量（auto）和静态局部变量（static）两种，形参只能是自动局部变量。
自动局部变量在变量定义的开始处加关键字auto修饰，但是auto可以省略。
局部变量在默认情况下就是自动局部变量，简称自变量。
例如：int a，b；相当于 auto int a，b；一个函数的自变量空间，从执行其定义语句时创建（如果是形参，就是在被实参初始化时创建），开始了生命周期，同时进入它的作用域。
当函数执行结束，其自变量空间被撤销，生命周期结束，也离开了作用域。
自变量只能由它所属的函数直接访问，这是自变量的作用域，即自变量的作用域范围是它所属的函数。
当该函数结束，或中途去调用其他函数，自变量就离开它的作用域。
一个自变量不在它的生命周期，肯定也不在它的作用域，反之不一定。
一个函数中途去调用另一个函数，这时它作为主调函数处于“ 中断 ”或“ 等待 ”状态，其自变量还在生命周期，但是离开了作用域，等到程序流程从被调函数返回，这个函数的自变量又回到作用域。
由上述过程可以推断：两个函数中的自变量如果同名，不会发生冲突，因为两个函数在调用过程中只有两种调用关系，一种是主调函数与被调函数的关系，另一种是前后被调用的关系。
在第一种关系中，一个是主调函数，另一个是被调函数，在执行被调函数时，主调函数的自变 量空间依然存在，系统只能为被调函数的自变量另外分配空间，这使得它们的自变量空间肯定不同，因此不会发生冲突。
在第二种关系中，两个函数同是被调函数，只是先后被调用，这时它们的自变量空间可能相同，因为前者占用的空间被系统撤销之后可以分配给后者，然而用的时间段不同，因此也不会发生冲突。

编辑推荐

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