

Started on 2002011920:23ww

程序设计实习，北京大学，信息科学技术学院郭炜，刘家瑛 *code: 代码 *question: 提出的问题 d

1 基础包括 c 等

1.1 函数指针

函数指针，指向函数起始地址 *code c0001 *code cpp0001

c 语言快速排序函数: void qsort(void *base, int nelem, unsigned int width, int(* pfCompare)(const void *, const void *)); 这里面 *pfCompare 就是函数指针起始地址，元素个数，每个元素大小，比较函数（自己编写）int 比较函数 (const void *elem1, const void *elem2); 比较函数编写，返回负整数会让 elem1 在前，返回正整数会让 2 在前，返回 0 则 1 和 2 不分前后。

1.2 命令行参数

2020012017:44, win+R 搜索 cmd, notepad t001.txt copy file1.txt file2.txt

int main(int argc, char * argv[])... argc, 参数个数, >=1; argv, 指针数组, argv[0] 指向的字符串是第一个命令行参数, 指向文件名, argv[i] 指向 i+1 个命令行参数打印所有命令行参数的程序 *code c0003.c.

1.3 位运算

in comment

1.4 引用和常量

in comment

1.5 动态内存分配

in comment 2020012822:19

1.6 内联函数和函数重载

in comment 2020012822:19

1.7 函数的缺省参数

in comment 2020012822:19

2 结构化程序设计

2.1 历史

in comment **code cpp0004 2020013016:34 大问题分解也就是模块化得到若干子问题。pascal 语言发明人 worth 提出程序 = 数据结构 + 算法。程序：理解、修改、差错、重用四大难点。分析问题所含事物及事物之间的关系，面向对象的程序 = 类 + 类。。。同一类事物的共同属性抽象为数据结构，行为抽象为函数。事物抽象成类，数据结构和操作数据结构的算法封装成类。class 包含 data 和 function

程序设计语言历史：1960 年算法描述语言，ALGOL60 1963 年，剑桥大学发展 CPL，1967 年，剑桥大学 BCPL，1967 年，Simula 67，第一个面向对象的语言，提出类、子类等概念 1970 年，贝尔实验室，B 语言，1971，Smalltalk 1973 年，C 语言，1983 年，C++，更适合面向对象。1995，JAVA，

常用 c++ 编译器：GCC，visual c++，dev c，eclipse，borand c++Builder

2.2 类和成员函数

2020013118:41 成员变量和成员函数 == != > < >= <= 需要重载后才可以进行对象之间的比较

对象名. 成员名指针-> 成员名

类成员可访问范围：private，成员函数内，意义在于强制限制只用成员函数访问该变量 public，任何地方 protected 省缺时默认为私有类的成员函数可以访问当前对象和同类其他对象的全部属性和函数，而成员函数外的地方只能访问该类对象的公有成员

2.2.1 内联和重载

【2020021617:49】内联函数 2 种写法：inline + 成员函数；整个函数体在类内部分

重载成员函数，注意默认参数设置时不要产生二义性；

2.2.2 构造函数

1) 名字与类名相同，可以有参数，不能有返回值。2) 用于变量初始化、成员变量赋值等。对象生成时调用构造函数，防止对象不初始化就应用导致的问题。3) 编译器生成的默认构造函数无参数，不做任何操作。4) 1 个类可有多个构造函数。

2.2.3 复制构造函数

把一个对象成员变量复制到另一个对象 1) 用初始化语句，一个对象初始化另一个对象，赋值语句不行；2) 函数有参数是类 A 的对象，调用函数时 A 的赋值构造函数被调用，来初始化这个

形参；

2.2.4 类型转换构造函数

只有 1 个参数，实现类型转换。

2.2.5 析构函数 (Destructor)

1) 名称 = 类名；2) 没有参数，没有返回值；3) 一个类最多有 1 个；4) 自动被调用，释放分配的空间。

2.3 static

1) static 成员变量和成员函数，是所有对象共享的；2) sizeof 不计算 static；3) 不作用于具体对象，不需要通过对象访问；4) 本质是全局变量、全局函数；5) 使相关全局变量和全局函数统一封装在类中，便于管理。如在矩形类中，定义所有矩形的总面积和矩形总个数两个 static。访问；6) static 函数中不能访问静态的成员变量和函数 7) 在统计生成对象总数时，记得要写复制构造函数构造时的记数。

2.4 成员对象和封闭类

成员对象：一个类的成员变量是另一个类的对象。封闭类 (Enclosing)：有成员对象的类。

1) 封闭类构造函数的初始化方法为：类名:: 构造函数 (参数表): 成员变量 i(参数表), 成员变量 j(参数表)..... 2) 封闭类对象生成时，先按成员对象的说明顺序生成成员对象的构造函数，再执行封闭类的构造函数；消亡时相反，类似栈的。

2.5 友元函数

1) friend 关键字，类的友元函数可以访问该类的私有成员；2) 一个类的成员函数（包括构造函数、析构函数）可以作为另一个类的友元；3) 类 a 中声明友元类 b 时，b 的成员函数可以访问 a 的私有成员，友元类之间的关系不能传递，不能继承。

2.6 this 指针

c++ 开始没有编译器，翻译成 c 来编译。翻译成 c 时，成员函数翻译成全局函数，全局函数的变量要增加一个指向对象的 this 指针。1) 作用是指向成员函数作用的对象；2) 静态成员函数不能用 this 指针，静态成员函数的真实参数个数就是所定义的个数。

2.7 const 常量

常量对象: `const Class a;`

常量成员函数: 1) 函数作用期间不能修改成员变量。所以也不能调用同类的非常量函数 (静态成员变量可以修改, 静态成员函数可以调用); 2) 函数说明后加 `const` 关键字。3) 名字和参数相同的 `const` 函数和非 `const` 函数, 算重载。

常引用: 1) 不修改其引用变量的值。2) 用对象的常引用防止对象被修改, 对象作为函数参数的时候, 该参数生成时调用复制构造函数, 效率低; 指针做参数比较难看的问题。

3 运算符重载

用 `c++` 提供的传统的运算符对抽象数据类型进行运算。已有的运算符赋予多重含义。一般的可以重载为成员函数, 或者普通函数。返回类型 `operator` 运算符 (形参表)... 编译时, 表达式调用运算符函数, 根据运算符的操作数查找实参的类型决定用哪个函数。

3.1 +-

3.2 赋值运算符 = 重载

两边类型可不匹配, 如 `int` 匹配 `complex`; `char*` 给字符串只能重载为成员函数, 普通函数不行。浅复制: 逐个字节复制, 如 `s1=s2`, `s1` 的指针是指向了 `s2`。产生 2 个问题, 有部分空间没有指针操作产生垃圾, 对象注销会一块内存注销 2 次出现错误。深复制:

3.3 运算符重载为友元函数

为了访问私有成员。如实现 `5+c`。

3.4 流插入运算符重载

为了实现连续赋值输出, 重载为: 这里的返回值是 `ostream` 的引用, 返回的 `this` 指针说明返回的是 `ostream` 的对象 (比如在这里是 `cout`), 也就是返回 `cout` 的引用。所以