

C++直播答疑课

潘祖江
清华大学计算机系

什么是内存泄漏？虚析构函数如何防止内存泄漏？

内存泄漏就是在程序中分配的内存没有正确地释放。

在类的派生体系内，将基类的析构函数声明为`virtual`类型，可以使派生类的析构函数在恰当的时刻被调用。

从而避免一些由于析构函数调用带来的内存泄漏问题

构造函数、析构函数有哪些特殊之处？

构造函数不能是虚函数，声明一个构造函数为虚函数是一个语法错误。用 `new` 运算符动态构造对象是，构造函数会被调用。

析构函数是唯一的，即没有入口参数也没有返回类型，因此不能重载；析构函数不能继承但是可以是虚函数；析构函数体中不能使用 `return` 语句返回值；对于 `new` 运算符动态构造的对象，使用 `delete` 运算符清除时会隐含调用析构函数。

LAB 8-1 复数加减乘除

题目描述求两个复数的加减乘除。

要求使用 `c++ class` 编写程序。

输入描述第一行两个 `double` 类型数，表示第一个复数的实部虚部

第二行两个 `double` 类型数，表示第二个复数的实部虚部

输出描述输出依次计算两个复数的加减乘除，一行一个结果

输出复数先输出实部，空格，然后是虚部

重载《运算符

```
friend ostream& operator<< (ostream &out, const Complex &c);
```

QUESTION:为什么重载输入输出运算符必须是非成员函数？

ANS:

C++中的输入输出本身就是C++对左操作符和右移操作符的重载，所以：

`cout << i;` (其中`cout`是`ostream`对象) 可翻译为 `cout.operator<<(i);`

在XXX类中利用类成员函数的形式重载输出操作符,该函数必须只有一个`ostream`引用类型的参数,使用该重载函数时:

`x.operator<<(cout);` (其中`cout`是`ostream`对象, `x`为XXX类型的对象) 即 `x<<cout;`

“`x<<cout;`”这种形式与常用形式不符，将会造成混淆。

如果利用非成员函数形式重载输入操作符，则使用时为：`operator<<(cout, x);` 也可以为 `cout << x;`

LAB 8-2 圆的周长和面积

题目描述求圆的周长和面积，已知圆类从`shape`抽象类继承。

输入描述输入圆的半径

输出描述输出圆的周长和面积

纯虚函数

- 纯虚函数是一个在基类中声明的虚函数，它在该基类中没有定义具体的函数体，要求各派生类根据实际需要定义自己的版本，纯虚函数的声明格式为：

`virtual 函数类型 函数名(参数表)=0;`

- 带有纯虚函数的类是抽象类。一个抽象类自身无法实例化，它是为了抽象和设计的目的而建立的，它的主要作用是通过它为一个类族建立一个公共的接口，使它们能够更有效地发挥多态特性。纯虚函数的函数体，要由派生类自己定义。

虚析构函数

在C++中，不能声明虚构造函数，但是可以声明虚析构函数。如果有可能通过基类指针调用派生类对象的析构函数（通过`delete`），并且被析构的对象具有重要的析构函数，就需要让基类的析构函数成为虚函数。虚析构函数的声明语法为：

```
virtual ~类名();
```

示例：通过虚析构函数防止内存泄漏问题

LAB 8-3 三角形还是长方形？

题目描述 在多态概念中，基类的指针既可以指向基类的对象，又可以指向派生类的对象。我们可以使用`dynamic_cast`类型转换操作符来判断当前指针（必须是多态类型）是否能够转换成为某个目的类型的指针。

同学们先查找`dynamic_cast`的使用说明（如http://en.wikipedia.org/wiki/Runtime_type_information#dynamic_cast），然后使用该类型转换操作符完成下面程序（该题无输入）。

函数`int getVertexCount(Shape * b)`计算`b`的顶点数目，若`b`指向`Shape`类型，返回值为0；若`b`指向`Triangle`类型，返回值为3；若`b`指向`Rectangle`类型，返回值为4。

DYNAMIC_CAST 知识点讲解

语法

`dynamic_cast < new_type > (expression)`

- new_type* - 指向完整类类型的指针、到完整类类型的引用，或指向（可选的 cv 限定）void 的指针
- expression* - 若 *new_type* 为引用，则为完整类类型的左值 (C++11 前) 泛左值 (C++11 起) 表达式，若 *new_type* 为指针，则为指向完整类类型的指针纯右值。

若转型成功，则 `dynamic_cast` 返回 *new_type* 类型的值。若转型失败且 *new_type* 是指针类型，则它返回该类型的空指针。若转型失败且 *new_type* 是引用类型，则它抛出匹配类型 `std::bad_cast` 处理块的异常。

选做题 8-1 打印螺旋数字正方形

题目描述打印螺旋数字正方形

输入描述输入一个整数 n ， $1 \leq n \leq 31$

输出描述输出数字正方形。

注意，每个数字占4个字符，右对齐，不全则补足空格；

选做题 8-2 字母菱形

题目描述打印字母菱形

输入描述输入一个大写字母与一个整数 n , $1 \leq n \leq 26$

输出描述输出如下高度为 $2n-1$ 的菱形;

注意: A的下一个字母是B, Z的下一个字母是A

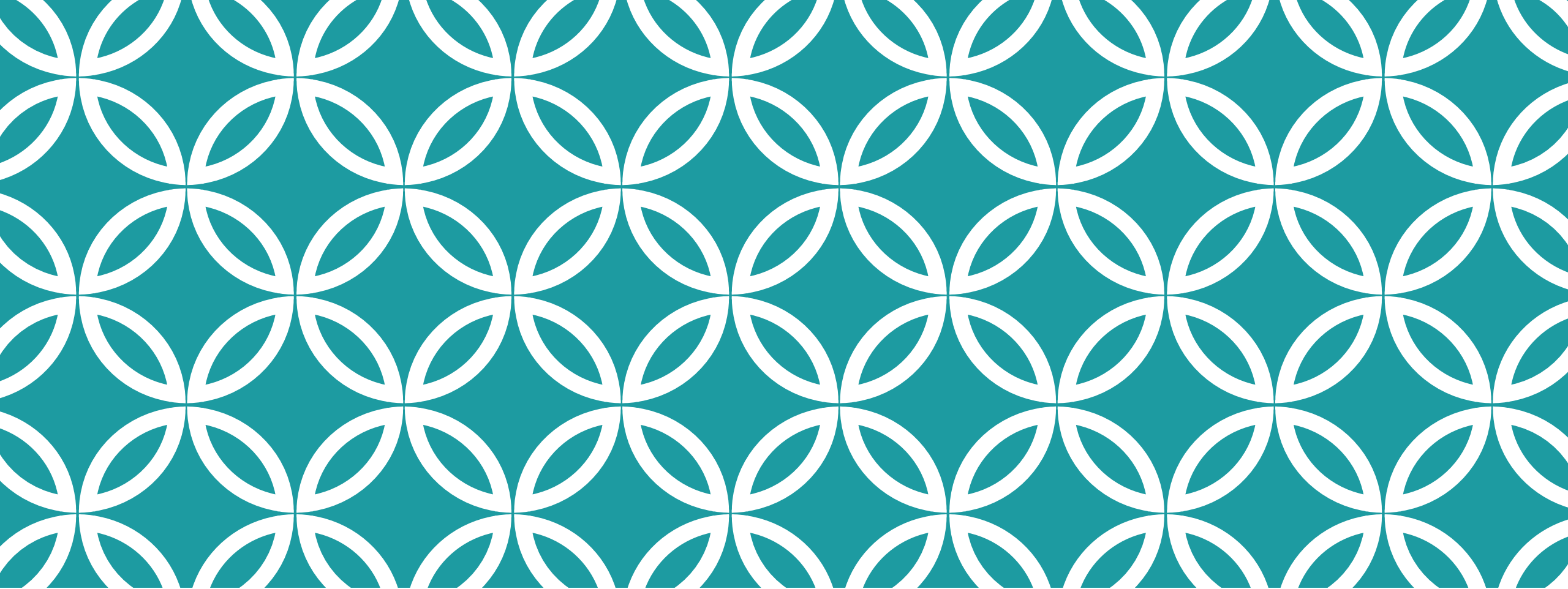
选做题 8-3 找单数

题目描述 对于一个长度为 N 的数组，有一个数只出现了一次，其他所有数都出现了2次，把这个出现一次的数找出来。输入 N 和 N 个数，输出出现一次的单数。

输入描述 第一行为数组长度

第二行为数组的元素

输出描述 要求的整数



LAB 9 |

LAB 9-1 数组求和

题目描述 编写一个模板函数 `getSum`，接收一个数组，返回该数组所有元素的和。部分代码已给出，请将代码填补完整。

输入描述 每个测例共 3 行，第一行为两个整数 $n, m (n > 1, m > 1)$ ，第二行为 n 个整数，用空格隔开，第三行为 m 个实数，用空格隔开。

输出描述 对每个测例输出两行，第一行为输入的 n 个整数的和，第二行为输入的 m 个实数的和。

函数模板和类模板

函数模板的定义形式是：

`template <class T>` 或 `template <typename T>`

类型名 函数名（参数表）

{函数体的定义}

所有函数模板的定义都是用关键字`template`开始的，该关键字之后是使用尖括号`<>`括起来的类型参数表。每一个类型参数之前都有关键字`class`或`typename`，这些类型参数代表的是类型，可以是内部类型或自定义类型。这样，类型参数就可以用来指定函数模板本身的形参类型、返回值类型，以及声明函数中的局部变量。函数模板中函数体的定义方式与定义其他函数类似。

函数模板和类模板 *CON'D*

类模板声明的语法形式是：

template <模板参数表>

class 类名

{类成员声明}

其中类成员声明的方法与普通类的定义几乎相同，只是在它的各个成员（数据成员和函数成员）中通常要用到模板的类型参数 *T*。

如果需要在类模板以外定义其成员函数，则要采用以下的形式：

template <模板参数表>

类型名 类名<*T*>::函数名（参数表）

LAB 9-2 折半查找

题目描述编写一个折半查找的模板函数`binSearch()`，接收一个数组，数组长度和要查找的元素`key`，按查找顺序输出查找过程中访问的所有元素下标。部分代码已给出，请将代码填补完整。

输入描述每个测例共 3 行，第一行为 4 个整数 $n, m (n > 1, m > 1)$ ，`key1, key2` (要查找的元素)。第二行为 n 个整数，已按由小到大排序，用空格隔开，第三行为 m 个实数，已按由小到大排序，用空格隔开。

输出描述对于每次查找，按查找顺序输出访问过的数组元素下标，每个下标占一行。

LAB 9-3 括号匹配

题目描述 栈的应用非常广泛。请先实现一个栈模板类（定义已在下面给出），然后利用这个栈类解决下面的问题：

给定一个字符串，长度小于**1000**，其中只包含左右括号和大小写英文字母。请编写程序判断输入的字符串里的左右括号是否全部是匹配的，匹配规则即从内到外左括号都与其右边距离最近的右括号匹配。如匹配，输出“**Yes**”，否则，输出“**No**”。

输入描述 每个测例输入一个字符串，长度小于 **1000**，大于 **0**。

输出描述 若输入字符串内的所有括号正确匹配，输出“**Yes**”，否则输出“**No**”。

LAB 9-4 翻转单词

题目描述 给定一个字符串，字符串中包含用空格隔开的单词。翻转字符串中所有单词的顺序。

输入描述 输入为一个长度不超过1000的字符串，字符串中有用空格隔开的单词。

输出描述 输出翻转单词顺序之后的字符串。

选做题 9-1 解析字符串

题目描述 给定一个字符串，返回解析后的结果。

解析规则为：将字符串中 `k[encode_string]` 部分展开成 `encode_string` 重复 `k` 次。

其中 `encode_string` 只会由小写字母构成。

输入描述 输入要解析的字符串，保证一定合法。

输出描述 输出解析后的字符串，占一行。

选做题 9-2 输出超一半数

题目描述 已知在一个`int`数组中，有一个数字出现超过了一半，找出该数字。
补充`find`函数，传入参数为数组首地址和数组长度，返回该数字。

方法1：排序

方法2：hash

方法3：单遍扫描数组即可

选做题 9-3 参加会议

题目描述 小明到学校之后看到需要参加的一系列会议时间安排，每个会议由起始时间和结束时间组成，所有的会议都已经按照开始时间从小到大排好顺序，如果开始时间相等，那么按照结束时间排序。

请判断小明能否参加所有的会议。

输入描述 输入一个整数 n ，表示有 n 个会议，其中 $0 < n < 1000$ 。

接下来 n 行，每行两个数字 a 和 b ， a 表示会议开始时间， b 表示会议结束时间。

输出描述 输出为一行，如果小明能参加所有的会议，则输出Yes，否则输出No