

# C++语言程序设计

---

## 第三章：类和使用

---

宋霜

哈尔滨工业大学（深圳）

机电工程与自动化学院

邮箱: **songshuang@hit.edu.cn**

# 第三章：类和使用

---

## 构造函数

- 处理对象的初始化
- 特殊的成员函数，不需要用户调用，建立对象时自动执行

# 第三章：类和使用

---

## 构造函数

- 处理对象的初始化
- 特殊的成员函数，不需要用户调用，建立对象时自动执行
- 构造函数没有返回值
- 不需要也不能被用户调用

# 第三章：类和使用

---

## 构造函数

- 处理对象的初始化
- 特殊的成员函数，不需要用户调用，建立对象时自动执行
- 函数名与类名相同
- 构造函数没有返回值
- 不需要也不能被用户调用
- 默认构造函数

# 第三章：类和使用

---

## 构造函数

- 带参数的构造函数
- 参数初始化列表
- 构造函数重载

# 第三章：类和使用对象

---

## 析构函数

- ~类名
- 处理对象空间的回收
- 特殊的成员函数，不需要用户调用，释放对象时自动执行
- 构造函数没有返回值
- 没有函数参数
- 默认析构函数

# 第三章：类和使用

---

## this指针

- 每个对象中的数据成员分别占有存储空间
- 不同对象都调用同一个成员函数的目标代码
- 指向本类对象的指针

# 第三章：类和使用

---

## const

- 类型名 const 变量名
- const 类型名 变量名



# 第三章：类和使用对象

---

## const

- 类型名 const 变量名
- const 类型名 变量名
- int const a
- const int a

# 第三章：类和使用

---

## const

- 类型名 const 变量名
- const 类型名 变量名
- int const a
- const int a
- int const \*p
- const int \*p
- int \* const p

# 第三章：类和使用

---

## const

- 类型名 const 变量名

- const 类型名 变量名

- int const a                      int const &a

- const int a                      const int &a

- int const \*p

- const int \*p

- int \* const p

# 第三章：类和使用

---

## const

- 类型名 const 变量名
- const 类型名 变量名

- |                 |                 |                 |
|-----------------|-----------------|-----------------|
| ■ int const a   | int const &a    |                 |
| ■ const int a   | const int &a    |                 |
| ■ int const *p  | f(int const a)  |                 |
| ■ const int *p  | f(int const &a) |                 |
| ■ int * const p | f(int const *a) | f(int* const a) |

# 第三章：类和使用

---

## **const**

- 初始化阶段
- 语句调用阶段

# 第三章：类和使用对象

---

## **const**

- 初始化阶段
  - const类型数据必须初始化
- 语句调用阶段

# 第三章：类和使用对象

---

## const

- 初始化阶段
  - const类型数据必须初始化
  - int const 与 int可以相互赋值
- 语句调用阶段

# 第三章：类和使用对象

---

## const

- 初始化阶段
  - const类型数据必须初始化
  - int const 与 int可以相互赋值
  - 引用 与 变量
- 语句调用阶段



# 第三章：类和使用对象

---

## const

- 初始化阶段
  - const类型数据必须初始化
  - int const 与 int可以相互赋值
  - 引用 与 变量
  - 指针 与 变量
- 语句调用阶段

# 第三章：类和对象的使用

---

## const

- 初始化阶段
  - const类型数据必须初始化
  - int const 与 int可以相互赋值
  - 引用 与 变量
  - 指针 与 变量
- 语句调用阶段
  - const修饰的变量不能改变值

# 第三章：类和使用

---

## 对象的常引用

- 作为函数参数

# 第三章：类和使用

---

## 对象的常引用

- 作为函数参数
- 拷贝（复制）构造函数

# 第三章：类和使用

---

## 常对象

- 类名 const 对象名(参数表)

# 第三章：类和对象的使用

---

## 常对象

- 类名 const 对象名(参数表)

## 常数据成员

- 通过构造函数的参数初始化列表来初始化

# 第三章：类和使用对象

---

## 常对象

- 类名 const 对象名(参数表)

## 常数据成员

- 通过构造函数的参数初始化列表来初始化

## 常成员函数

- 类型名 函数名（参数表） const {}
- 不能修改数据成员，不能调用非常成员函数
- 常对象只能调用常成员函数

# 第三章：类和使用

---

## 对象的动态建立和删除

- `vector_ss *p=new vector_ss;`
- `p->display();`
- `delete p;`



# 第三章：类和使用对象

---

## 静态成员

- static 修饰的成员
- 静态成员变量
- 静态成员函数
  - 用来引用静态数据成员

# 第三章：类和对象的使用

## 静态成员

- static 修饰的成员
- 静态成员变量
- 静态成员函数
  - 用来引用静态数据成员

	静态成员函数	普通成员函数
所有对象共享	Yes	Yes
隐含 this 指针	No	Yes
访问普通成员变量（函数）	No	Yes
访问静态成员变量（函数）	Yes	Yes
通过类名直接调用	Yes	No
通过对象名直接调用	Yes	Yes

# 第三章：类和使用

---

## 友元函数

- friend修饰函数
- 可以访问对象的私有数据
- 可以是另一个类中的成员函数

# 第三章：类和使用对象

---

## 友元函数

- friend修饰函数
- 可以访问对象的私有数据
- 可以是另一个类中的成员函数

## 友元类

- friend 类名
- 友元类中任意一个函数都可以访问类的成员
- 单向，不能传递

# 第三章：类对象的使用

---

## 类模板

- 定义
  - `template<class typename>`
  - `class A`
  - `{};`
- 使用
  - `A<int> a;`