

第8章 取其精华 发挥优势—继承

1、模拟人的行走、听、说、写

2、为什么需要继承

3、派生类的定义

4、基类与派生类

5、三种继承方式

6、派生类的构造与析构函数

7、点、圆、圆柱体继承设计

8、从U盘到MP3继承设计

派生类构造与析构函数

- ▶ 基类的构造函数与析构函数不能被继承
- ▶ 派生类构造函数的一般形式为：

```
派生类名::派生类名(参数总表): 基类名1(参数表1), ..., 基类名n(参数表n),  
    内嵌对象名1 (对象参数表1) , ..., 内嵌对象名m (对象参数表m)  
{  
    派生类新增加成员的初始化;  
}
```

基类Person类的定义

```
#include<iostream>
#include<string.h>
using namespace std;
class Person
{
    char Name[10];           //姓名
    int    Age;              //年龄
public:
    Person(char* name,int age)
    {
        strcpy(Name, name);
        Age = age;
        cout<<"constructor of person"<<Name<<endl;
    }
    ~Person() { cout<<"deconstrutor of person"<<Name<<endl;}
};
```

派生类构造与析构函数执行测试

```
class Employee : public Person
```

```
{
```

```
    char    Dept[20];
```

```
    Person  Leader;
```

```
public:
```

```
    Employee(char *name, int age, char *dept, char *leader)
```

```
        : Person(name,age) , Leader(leader)
```

```
    {        strcpy(Dept, dept);
```

```
        cout<<"constructor of Employee"<<endl;
```

```
    }
```

```
    ~Employee() { cout<<"deconstrucor of Employee"<<endl; }
```

```
};
```

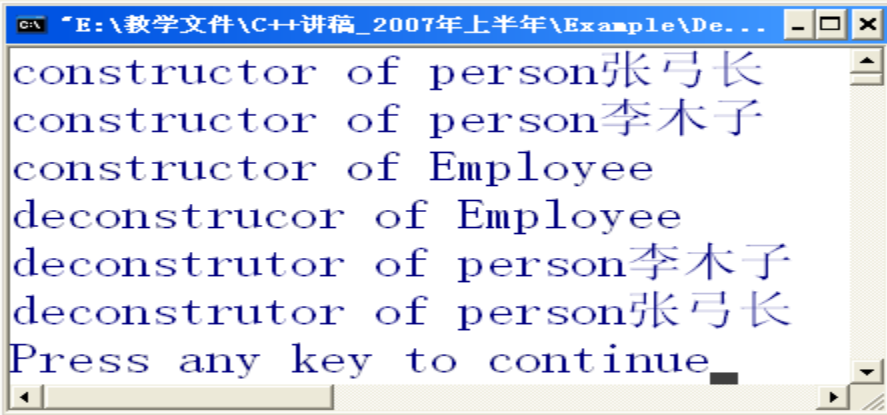
```
int main()
```

```
{
```

```
    Employee emp("张弓长",40,"人事处","李木子",36);
```

```
    return 0;
```

```
}
```



The screenshot shows a Windows-style application window titled "E:\教学文件\C++讲稿_2007年上半年\Example\De...". The window contains the following output text:

```
constructor of person张弓长
constructor of person李木子
constructor of Employee
deconstrucor of Employee
deconstrutor of person李木子
deconstrutor of person张弓长
Press any key to continue
```

派生类构造函数的执行次序

- **首先，调用基类构造函数，调用顺序按照它们被继承时声明的基类名顺序执行。**
 - **其次，调用内嵌对象构造函数，调用次序按各个对象在派生类内声明的顺序。**
 - **最后，执行派生类构造函数体中的内容。**
-

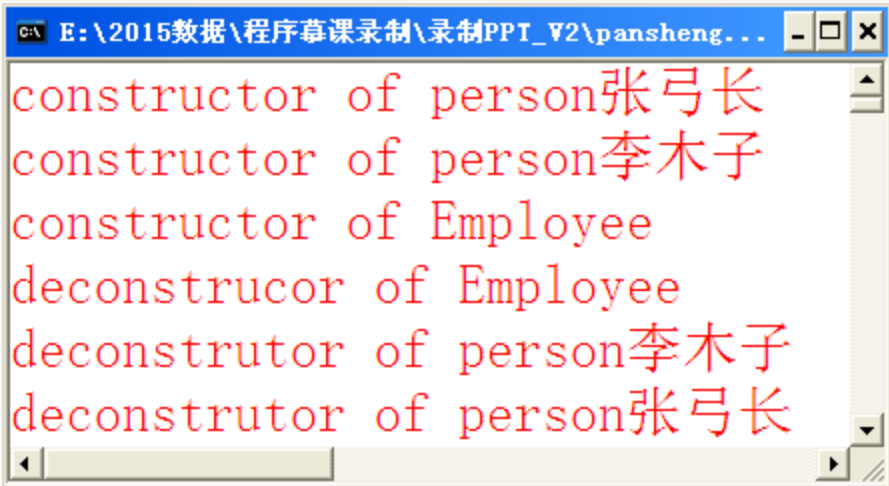
派生类析构函数的执行次序

❑ 派生类析构函数执行过程恰与构造函数执行过程相反。

❑ 首先执行派生类析构函数

❑ 然后执行内嵌对象的析

❑ 最后执行基类析构函数



```
E:\2015数据\程序课录制\录制PPT_V2\pansheng...  
constructor of person张弓长  
constructor of person李木子  
constructor of Employee  
deconstrucor of Employee  
deconstrutor of person李木子  
deconstrutor of person张弓长
```