

C和C++结构体的区别

1. C的结构体内不允许有函数存在，C++允许有内部成员函数，且允许该函数是虚函数。所以C的结构体是没有构造函数、析构函数、和this指针的。
2. C的结构体对内部成员变量的访问权限只能是public，而C++允许public,protected,private三种。
3. C语言的结构体是不可以继承的，C++的结构体是可以从其他的结构体或者类继承过来的。
4. 在C中定义一个结构体类型要用typedef，如下：

```
typedef struct Complex{  
    int read;  
    int image;  
}Complex;
```

那么，在说明Complex变量的时候可以这样写

```
Complex complex;
```

但是如果没typedef就必须用

```
struct Complex complex;
```

来声明。这里的Complex实际上就是struct Complex的别名。另外这里也可以不写Complex（于是也不能struct Complex complex;了）

```
typedef struct{  
    int read;  
    int image;  
}Complex;
```

但在c++里很简单，直接

```
struct Complex{
    int read;
    int image;
};
```

于是就定义了结构体类型Complex，声明变量时直接Complex complex;

5. 在c++中如果用typedef的话，又会造成区别：

```
struct Complex1{
    int read;
    int image;
}complex;//complex是一个变量
```

```
typedef struct Complex2{
    int read;
    int image;
}Complex2;//Complex2是一个结构体类型
```

使用时可以直接访问complex.read，但是Complex2则必须先Complex2 complex2;然后complex2.read = 1;

6. 掌握上面两条就可以了，不过最后我们探讨个没多大关系的问题。如果在c程序中我们写：

```
typedef struct{
    int num;
    int age;
}aaa,bbb,ccc;
```

这算什么呢？我个人观察编译器（VC6）的理解，这相当于

```
typedef struct{
    int num;
    int age;
}aaa ;
typedef aaa bbb;
```

```
typedef aaa ccc;
```

也就是说aaa,bbb,ccc三者都是结构体类型。声明变量时用任何一个都可以,在c++中也是如此。但是你要注意的是这个在c++中如果写掉了typedef关键字,那么aaa,bbb,ccc将是截然不同的三个对象。

本博客文章除特别声明,全部都是原创!
原创文章版权归过往记忆大数据([过往记忆](#))所有,未经许可不得转载。
本文链接:【】()