

挑战30天C 入门极限：c类的多重继承与虚拟继承 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/134/2021_2022__E6_8C_91_E6_88_9830_E5_A4_c97_134506.htm

在过去的学习中，我们始终接触的单个类的继承，但是在现实生活中，一些新事物往往会拥有两个或者两个以上事物的属性，为了解决这个问题，C引入了多重继承的概念，C允许为一个派生类指定多个基类，这样的继承结构被称做多重继承。举个例子，交通工具类可以派生出汽车和船连个子类，但拥有汽车和船共同特性水陆两用汽车就必须继承来自汽车类与船类的共同属性。由此我们不难想出如下的图例与代码：当一个派生类要使用多重继承的时候，必须在派生类名和冒号之后列出所有基类的类名，并用逗好分隔。//程序作者:管宁 //站

点:www.cndev-lab.com //所有稿件均有版权,如要转载,请务必著名出处和作者 #include iostream>usingnamespacestd. classVehicle

```
{ public: Vehicle(intweight =0) { Vehicle::weight =weight. }  
voidSetWeight(intweight) { coutVehicle::weight =weight. }  
virtualvoidShowMe() =0. protected: intweight. }.
```

```
classCar:publicVehicle//汽车 { public:  
Car(intweight=0,intaird=0):Vehicle(weight) { Car::aird =aird. }  
voidShowMe() { cout} protected: intaird. }.
```

```
classBoat:publicVehicle//船 { public:  
Boat(intweight=0,floattonnage=0):Vehicle(weight) { Boat::tonnage  
=tonnage. } voidShowMe() { cout} protected: floattonnage. }.
```

```
classAmphibianCar:publicCar,publicBoat//水陆两用汽车,多重继承的体现 { public: AmphibianCar(intweight,intaird,floattonnage)
```

```
:Vehicle(weight),Car(weight,aird),Boat(weight,tonnage) //多重继承  
要注意调用基类构造函数 { } voidShowMe() { cout } }.  
intmain() { AmphibianCar a(4,200,1.35f).//错误 a.SetWeight(3).//  
错误 system("pause"). } 上面的代码从表面看，看不出有明显的  
语发错误，但是它是不能够通过编译的。这有是为什么呢？  
这是由于多重继承带来的继承的模糊性带来的问题。先看如  
下的图示：在图中深红色标记出来的地方正是主要问题所在  
，水陆两用汽车类继承了来自Car类与Boat类的属性与方法  
，Car类与Boat类同为AmphibianCar类的基类，在内存分配  
上AmphibianCar获得了来自两个类的SetWeight()成员函数，  
当我们调用a.SetWeight(3)的时候计算机不知道如何选择分别  
属于两个基类的被重复拥有了的类成员函数SetWeight()。由  
于这种模糊问题的存在同样也导致了AmphibianCar  
a(4,200,1.35f).执行失败，系统会产生Vehicle ”不是基或成员的  
错误。 100Test 下载频道开通，各类考试题目直接下载。详细  
请访问 www.100test.com
```