

OOinC(2) : C语言的多态实现方法 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/225/2021\\_2022\\_OOinC\\_2\\_\\_EF\\_BC\\_c97\\_225712.htm](https://www.100test.com/kao_ti2020/225/2021_2022_OOinC_2__EF_BC_c97_225712.htm) 计算机等级考试训练软件《百宝箱》相信很多人都看过设计模式方面的书，大家有什么体会呢

？Bridge，Proxy，Factory这些设计模式都是基于抽象类的。使用抽象对象是这里的一个核心。其实我觉得框架化编程的一个核心问题是抽象，用抽象的对象构建程序的主体框架，这是面向对象编程的普遍思想。用抽象构建骨架，再加上多态就形成了一个完整的程序。由于C++语言本身实现了继承和多态，使用这样的编程理念（理念啥意思？跟个风，嘿嘿）在C++中是十分普遍的现象，可以说Virtual（多态）是VC的灵魂。但是，使用C语言的我们都快把这个多态忘光光了。我常听见前辈说，类？多态？我们用的是C，把这些忘了吧。很不幸的是，我是一个固执的人。这么好的东西，为啥不用呢。很高兴的，在最近的一些纯C代码中，我看见了C中的多态！下面且听我慢慢道来。

1.VC中的Interface是什么Interface：中文解释是接口，其实它表示的是一个纯虚类。不过我所要说的是，在VC中的Interface其实就是struct，查找Interface的定义，你可以发现有这样的宏定义：`#ifndef Interface #define Interface struct #endif`而且，实际上在VC中，如果一个类有Virtual的函数，则类里面会有vtable，它实际上是一个虚函数列表。实际上C++是从C发展而来的，它不过是在语言级别上支持了很多新功能，在C语言中，我们也可以使用这样的功能，前提是我们不得不自己实现。

2.C中如何实现纯虚类（我称它为纯虚结构）比较前面，相信大家已经豁

然开朗了。使用struct组合函数指针就可以实现纯虚类。例子：  
：typedef struct { void ( \*Foo1 ) ( ) ; char ( \*Foo2 ) ( )  
； char\* ( \*Foo3 ) ( char\* st ) ; }MyVirtualInterface；这样假设我们在主体框架中要使用桥模式。（我们的主类是DoMyAct，接口具体实现类是Act1，Act2）下面我将依次介绍这些“类”。（C中的“类”在前面有说明，这里换了一个，是使用早期的数组的办法）主类DoMyAct：主类中含有MyVirtualInterface\* m\_pInterface；主类有下函数：  
： DoMyAct\_SetInterface ( MyVirtualInterface\* pInterface ) { m\_pInterface= pInterface； } DoMyAct\_Do ( ) { if ( m\_pInterface==NULL ) return； m\_pInterface->Foo1 ( ) ； c=m\_pInterface->Foo2 ( ) ； }子类Act1：实现虚结构，含有MyVirtualInterface st[MAX]；有以下函数：  
： MyVirtualInterface\* Act1\_CreatInterface ( ) { index=FindValid ( ) //对象池或者使用Malloc！应该留在外面申请，实例化if ( index==-1 ) return NULL； St[index].Foo1=Act1\_Foo1； // Act1\_Foo1要在下面具体实现St[index].Foo2=Act1\_Foo2 ； St[index].Foo3=Act1\_Foo3； Return &st [index]； }子类Act2同上。 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](http://www.100test.com)