第五章课后题心得体会

孙浩然 202031906076

Copyright ©2021-2099 HaoranSun. All rights reserved

第五章学习了多态性，这一张的内容主要与重载相关，并且与之前内容联系比较紧密；在面向对象方法中，多态性就是不同对象收到相同消息，产生不同的行为，比如重载。在C++程序设计中，多态性是指用一个名字定义不同的函数，这些函数执行不同但又类似的操作，这样就可以用同一个函数名调用不同内容的函数。换言之，可以用同样的接口访问功能不同的函数，从而实现“一个接口，多种方法”。而多态有两种表现形式：静态多态性（通过一般的函数重载来实现）、动态多态性（通过虚函数来实现）。静态多态性比较简单，主要动态多态性比较难理解。动态多态性有两个条件：

1. 在基类中必须使用虚函数、纯虚函数
2. 调用函数时要使用基类的指针或引用

对于普通函数的处理：一个特定的函数都会映射到特定的代码，无论是编译阶段还是连接阶段，编译器都能计算出这个函数的地址，调用即可。虚函数的处理：被调用的函数不仅依据调用的特定函数，还依据调用的对象的种类。通常是由虚函数表(vtable)来实现的。而虚函数表既有继承性又有多态性，每个派生类的vtable继承了它各个基类的vtable，如果基类vtable中包含某一项，则其派生类的vtable中也将包含同样的一项，但是两项的值可能不同。如果派生类重载(override)了该项对应的虚函数，则派生类vtable的该项指向重载后的虚函数，没有重载的话，则沿用基类的值。

Copyright ©2021-2099 HaoranSun. All rights reserved