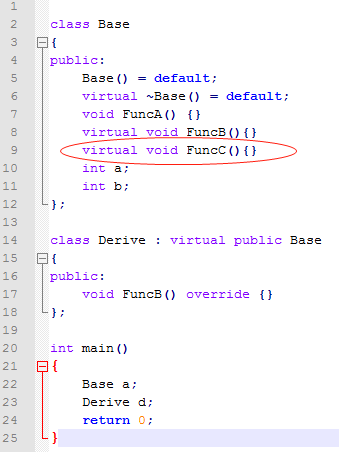
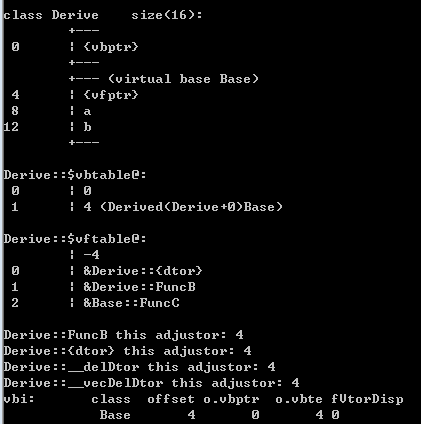
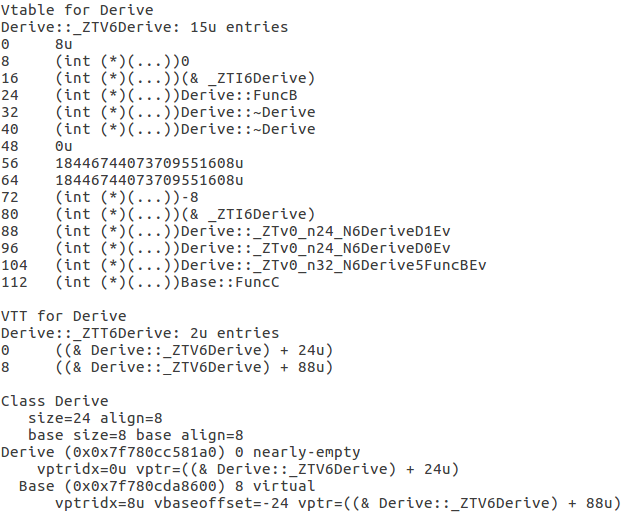
**虚继承带未覆盖函数的对象布局**



Windows平台下cl编译器：

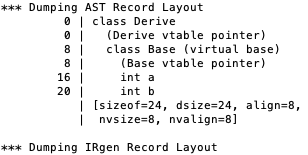


Linux平台下gcc编译器：



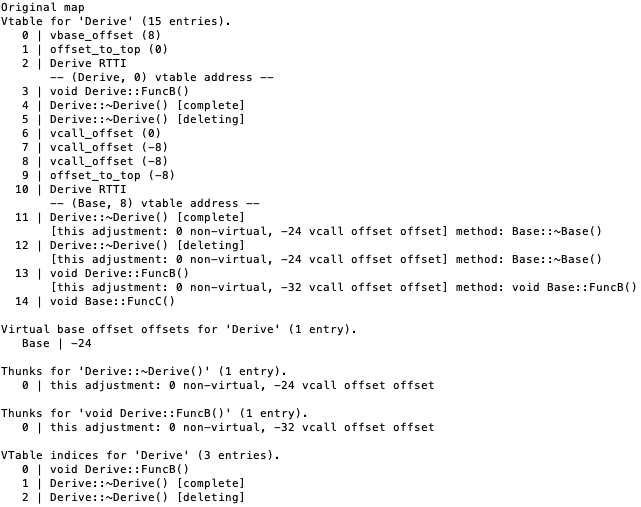
Mac平台下clang编译器：

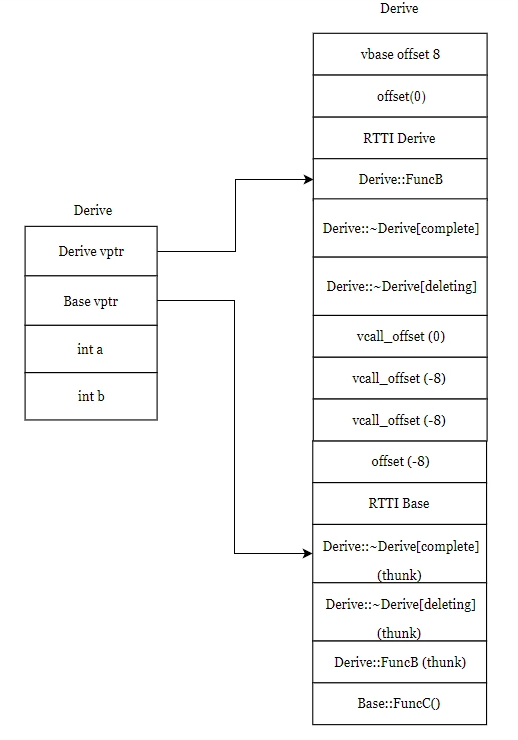
子类对象布局



和第8个例子相同，普通单继承下子类和基类共用一个虚表地址，而在虚继承下，子类和虚基类分别有一个虚表地址指针，两个指针大小总和为16，在加上a和b的大小8，为24。

虚函数表布局





**vbase\_offset(8)**：对象在对象布局中所指向虚基类虚函数表的指针地址的偏移量。

**vcall\_offset(-8)**：当**虚基类**Base的引用或指针base实际接受的是Derive类型的对象，执行base->FunB()时候，由于FuncB()已经被重写，而此时的this指针指向的是Base类型的对象，需要对this指针进行调整，就是vcall\_offset(-8)，所以this指针是向上调整了8字节，之后调用FuncB()，就调用到了被重写后的FuncB()函数。

**vcall\_offset(0)**：当Base的引用或指针base实际接受的是Derive类型的对象，执行base->FunC()时候，由于FuncC()没有重写，所以不需要对this指针进行调整，就是vcall\_offset(0);