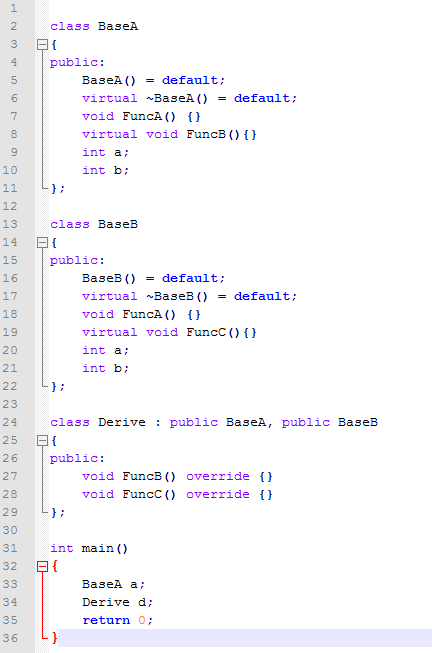
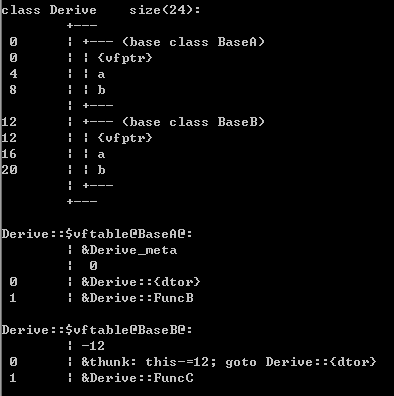
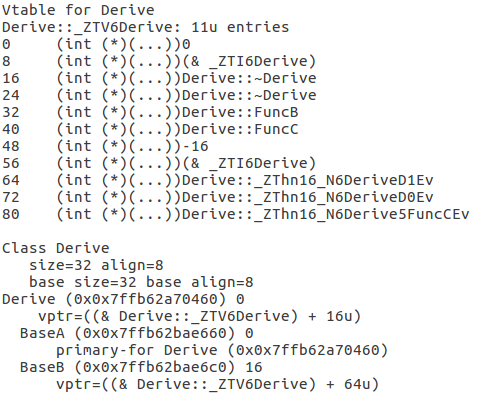
**多继承下不含覆盖函数的类对象的布局**



Windows平台下cl编译器：

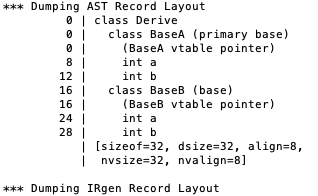


Linux平台下gcc编译器：

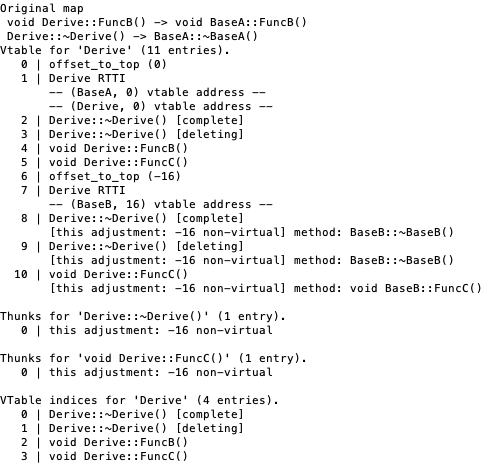


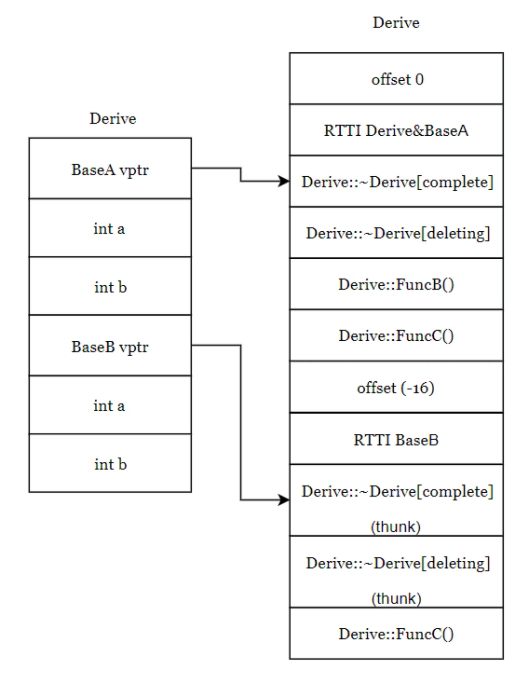
Mac平台下clang编译器：

子类对象布局



虚函数表布局





offset\_to\_top(0)：表示当前这个虚函数（BaseA,Derive）地址距离对象顶部地址的偏移量，因为对象的头部就是虚函数表的指针，所以偏移量为0。

注意这里的RTTI中有了两项，表示BaseA和Derive的虚表地址是相同的，BaseA类里面的虚函数和Derive类里的虚函数都在这个链条下。截至到offset\_to\_top(-16)之前都是BaseA和Derive的虚函数表。

offset\_to\_top(-16)：表示当前这个虚函数表(BaseB)地址距离顶部地址的偏移量，因为对象的头部就是虚函数的指针，所以偏移量为-16，当基类BaseB的引用或者指针base实际接受的是Derive类的对象，执行base->FuncC()的时候，由于FuncC()已经被重写，而此时的this指针执行的是BaseB类型的对象，需要对this指针进行调整，就是offset\_to\_top(-16)，之后条用FuncC()，就调用到被重写后Derive虚函数表中的FuncC()函数。这些带adjustment标记的函数都是需要进行指针调整的。