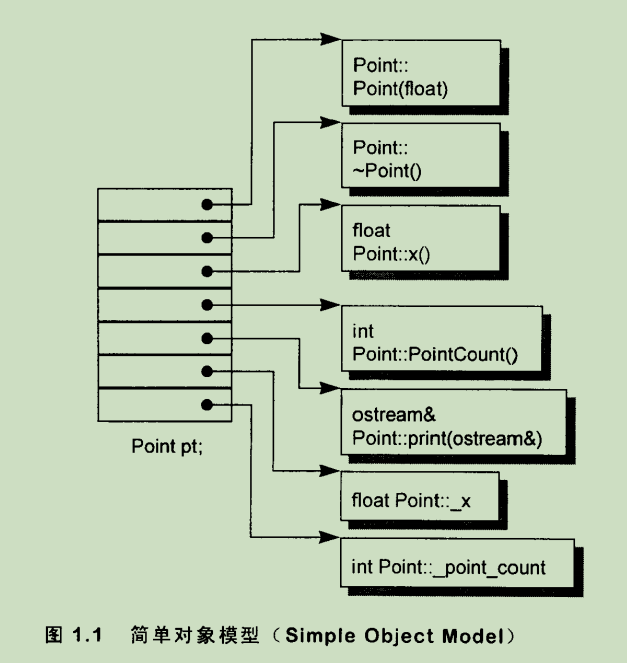
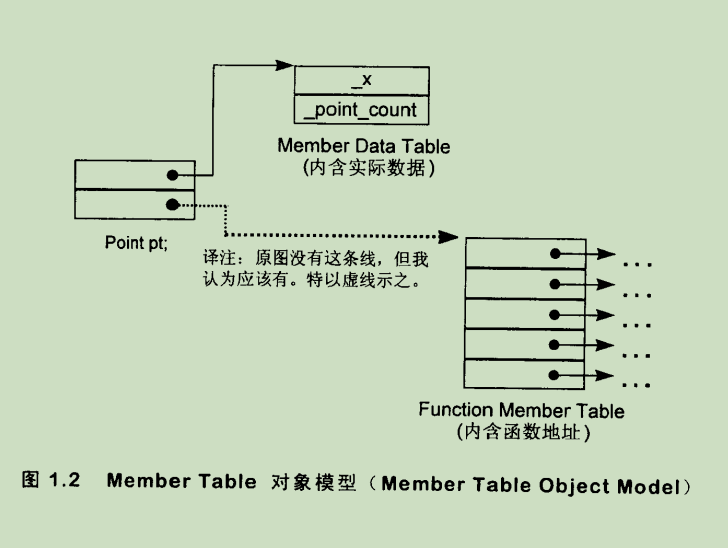
1. 关于对象

\*复习C++多态机制， slice

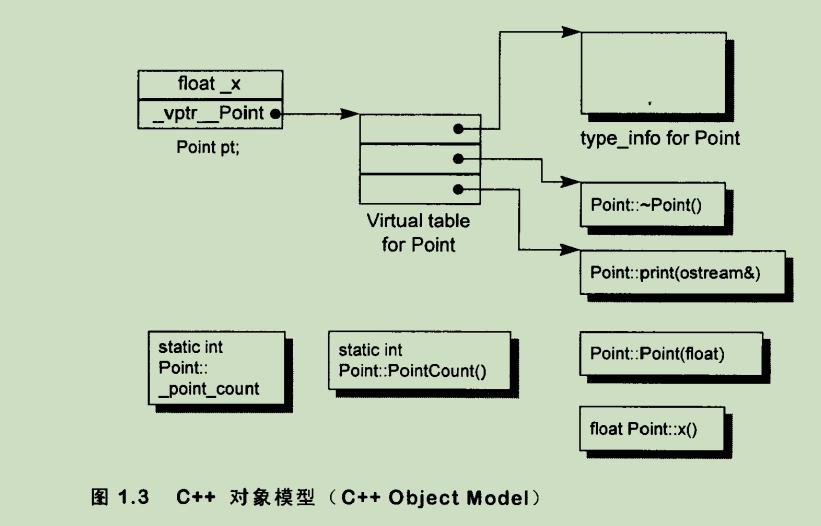
\*简单对象模型



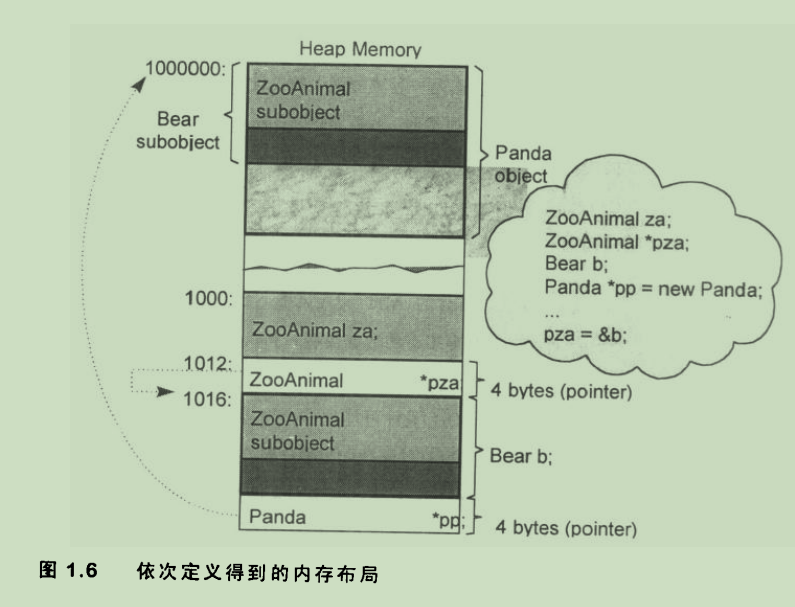
\*表格驱动模型(A Table-driven Object Model)



\*C++ 对象模型



\*



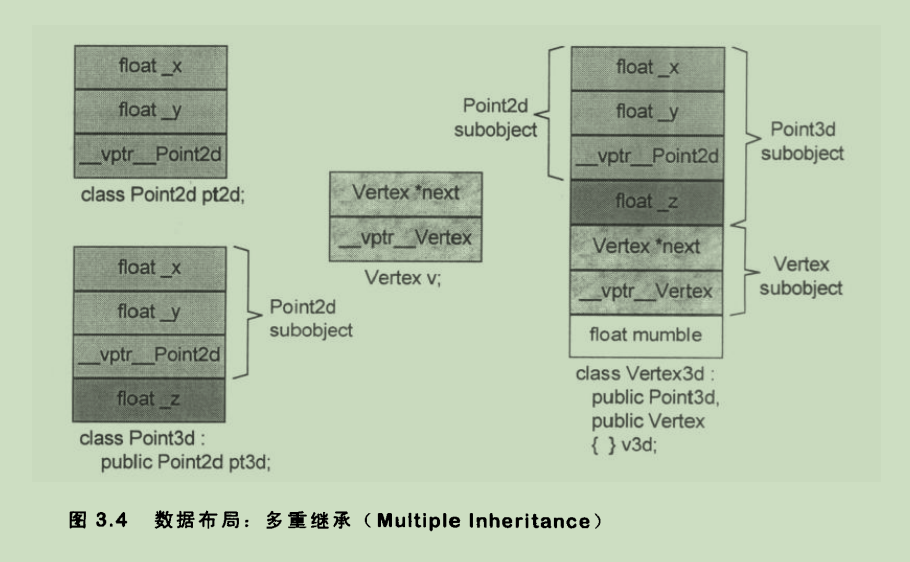
1. 构造函数语意学

\*NRV优化：

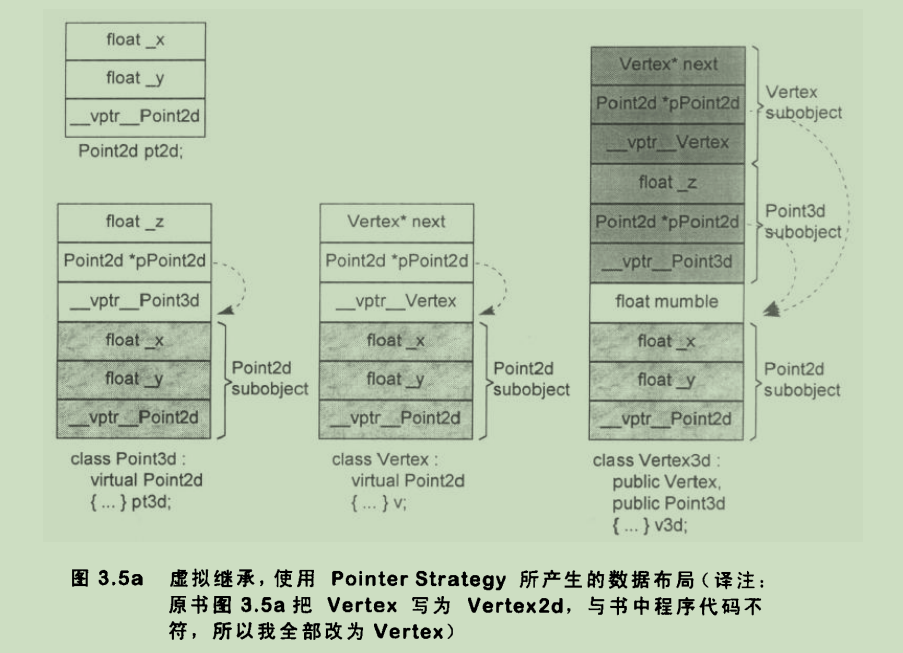
1. Data 语意学

\*static data member 存放在程序的 data segment中，和个别的class objects无关

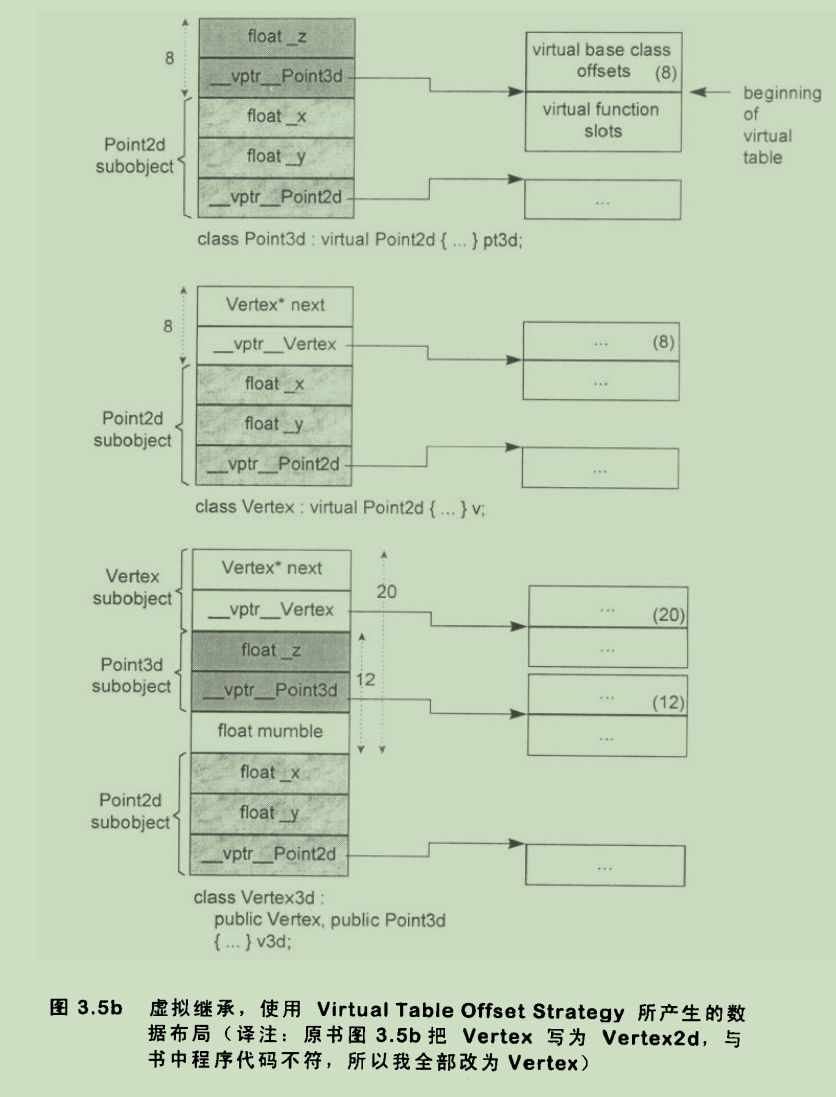
\*多重继承布局



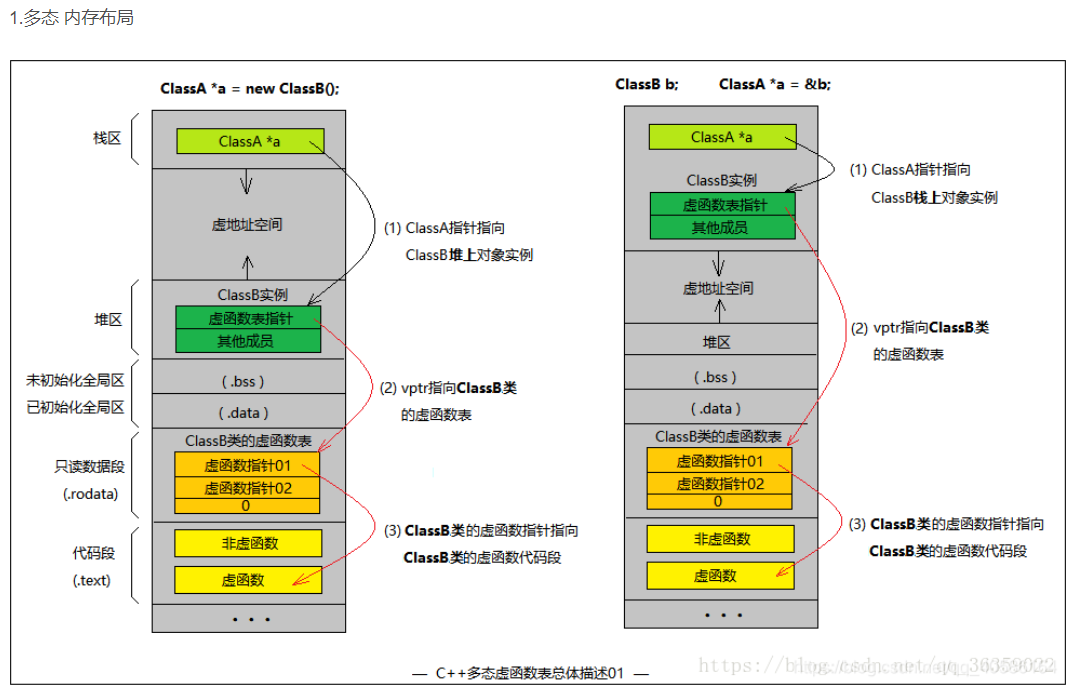
\*virtual 继承 （Cfront 实现）



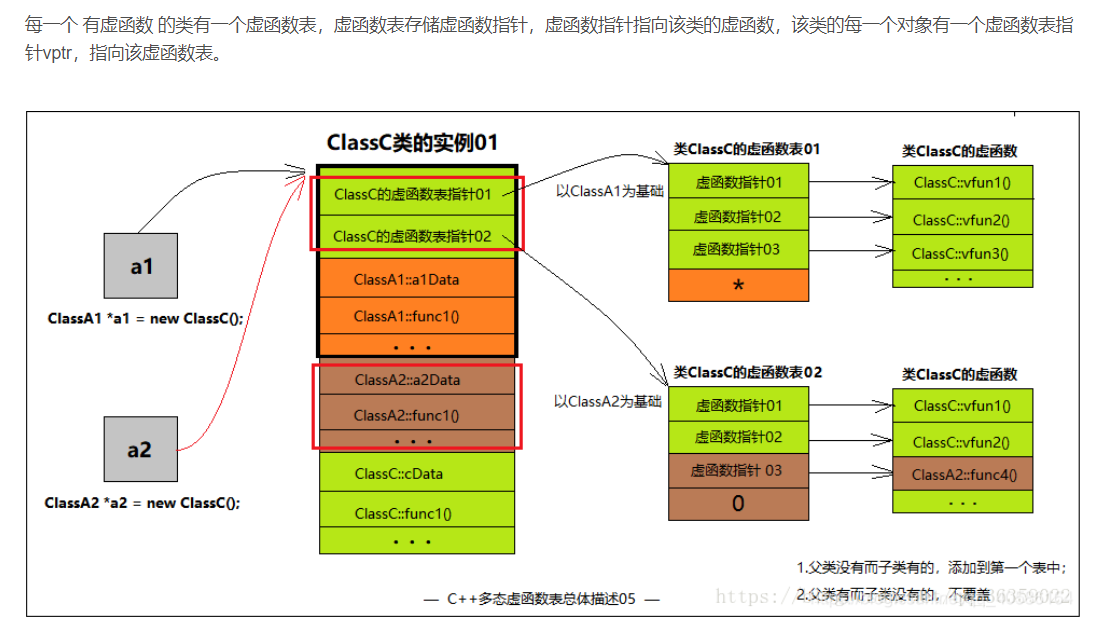
\*Foundation项目中的实现



\*\*\*\*\*\*



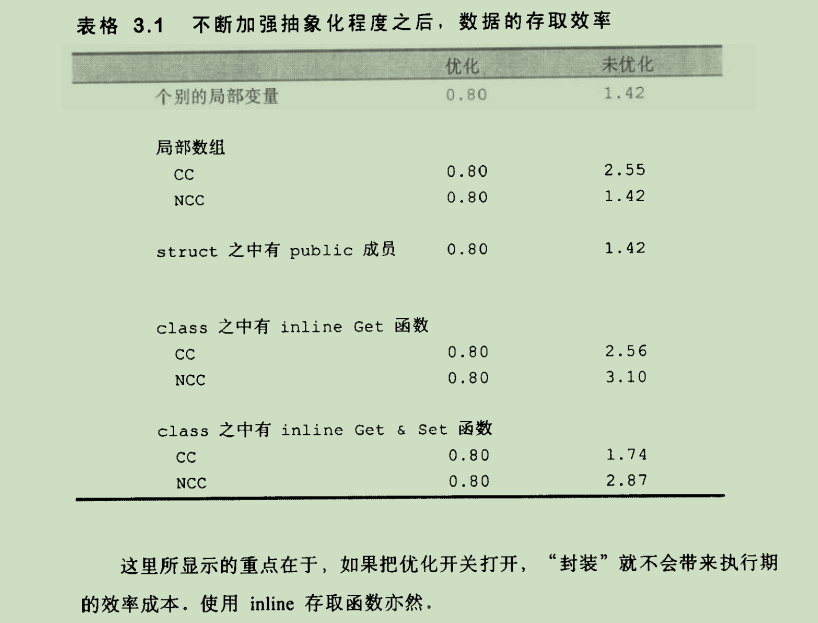
\*\*\*\*\*



\*virtual base class 最有效的一种运用形式就是：一个抽象的virtual base class，没有任何data members。

\*

\*优化相关





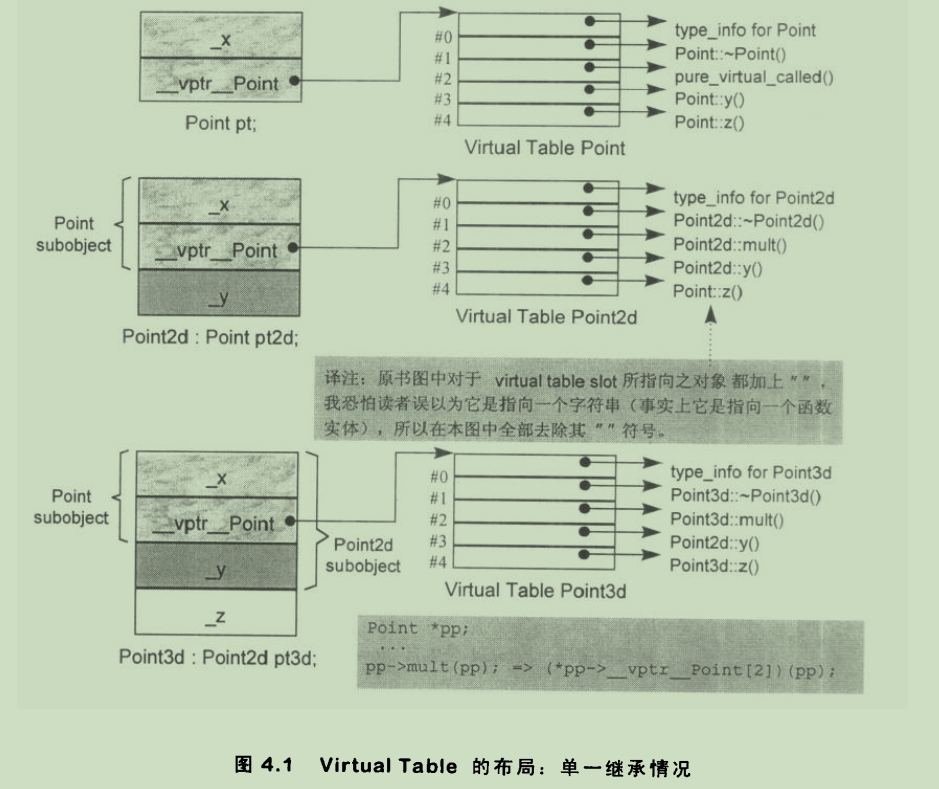
1. Function 语意学

\*member function 被转化为nonmember的形式，转化步骤为：

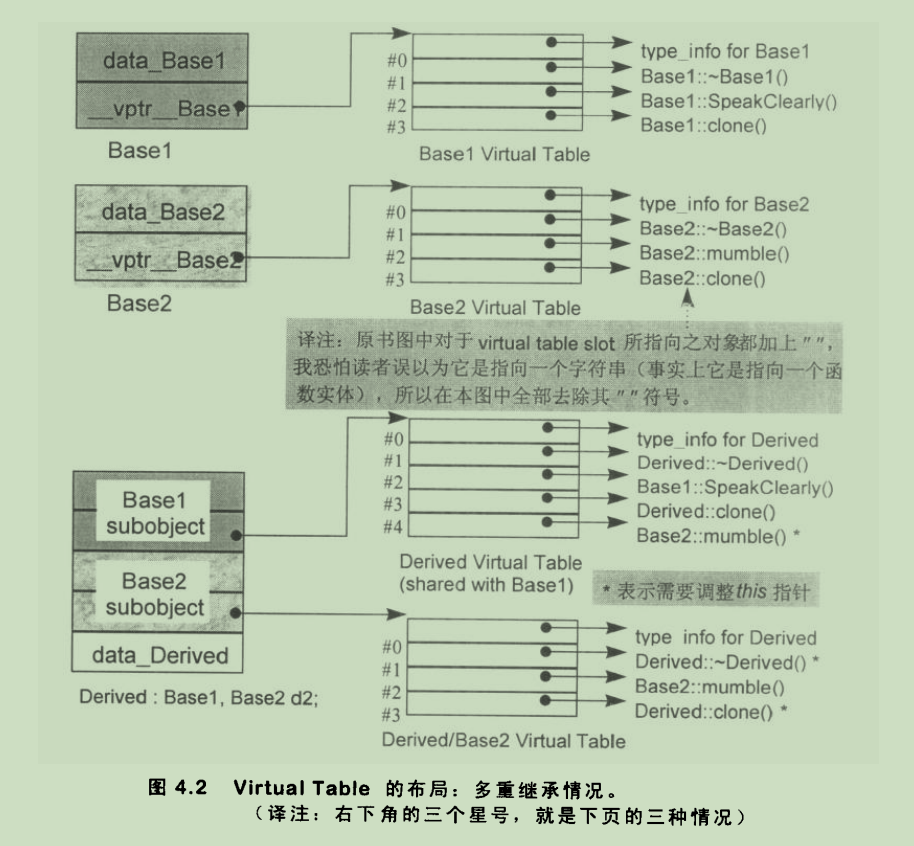
+1. Point3d Point3d::magnitude(Point3d\* const \_\_this);

+2.magnitude\_\_7Point3dFv(&obj);

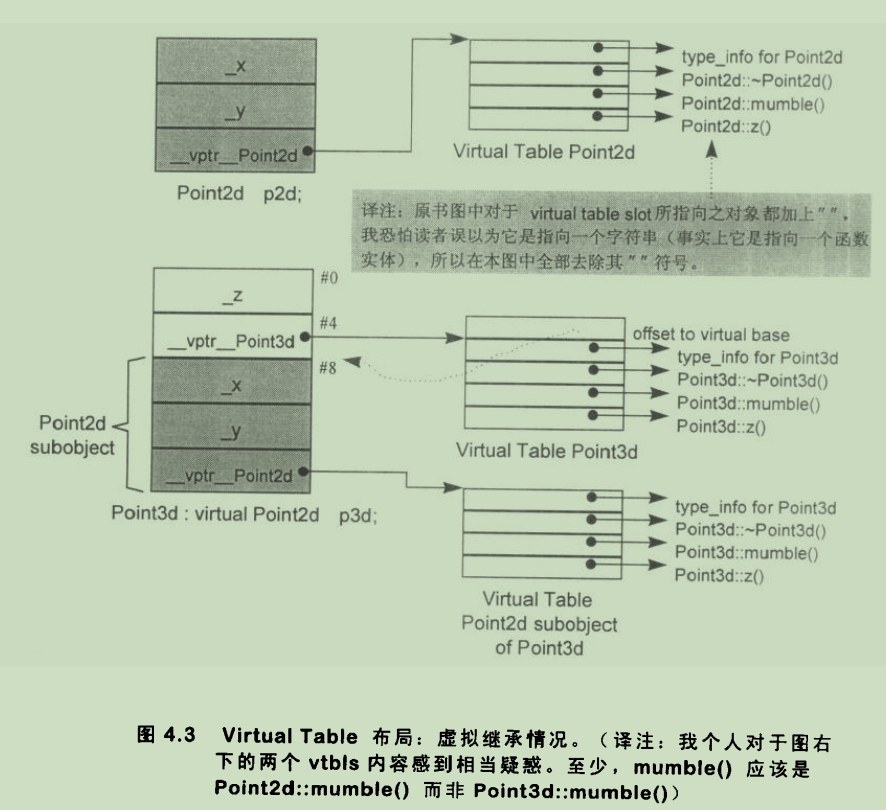
\*\*\*\*\*\*\*\*\*



\*\*\*\*\*\*\*\*\*\*\*\*\*\*



\*\*\*\*\*\*\*\*\*\*\*\*\*

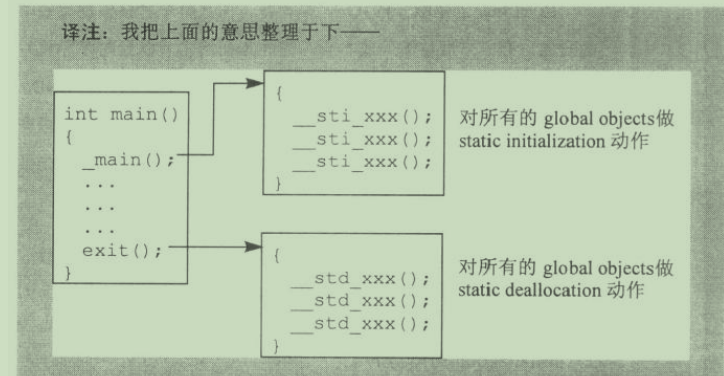


1. 构造，析构，拷贝语意学

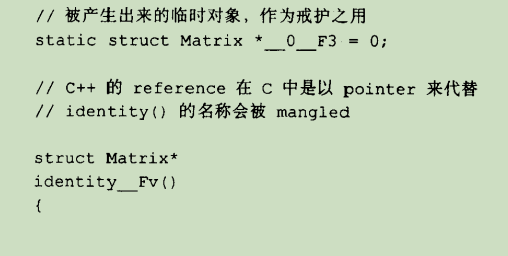
\*

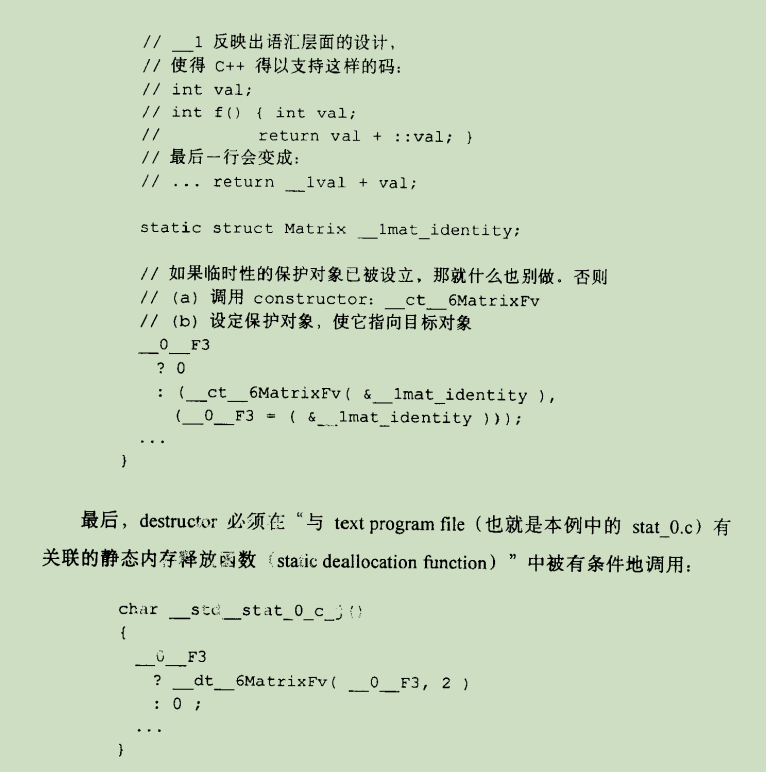
1. 执行期语意学

\*

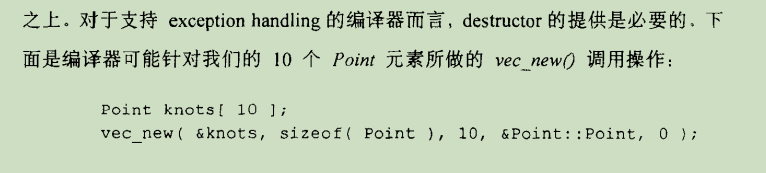


\*\*\*\*析构函数调用的操作





\*\*\*对数组的操作



1. 站在对象模型的类端

\*7.1 未读

\*7.2 未读

\*7.3 dynamic\_cast:检查基类的类型，非继承基类的转换会返回null，但是C类型转换是不检查继承关系。

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*完结\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*