C++程序设计期末考试-参考PPT节选补充 2022.11.25

第7章

使用名字空间特定成员时，会将该成员定义加入当前作用域，因此就不能再定义和该成员同名的标识符；而引用名字空间与引用名字空间成员不同，不会将其成员定义加入到当前作用域，可以在当前作用域定义和名字空间中标识符同名的标识符。当引用的名字空间成员和函数外定义的变量同名时，用“::变量名”访问外部变量，用“名字空间名称::变量名”访问名字空间成员。

如果类A的实例函数成员被声明为类B的成员友元，则这种友元称为**实例成员友元**。如果类A的静态函数成员被声明为类B的成员友元，则这种友元称为**静态成员友元**。

友元（类）关系不能传递、交换。一个普通函数可以定义为多个类的普通友元，友元函数的参数也可以缺省和省略。

内联的友元函数的存储类默认为static，作用域局限于当前代码文件。全局main的作用域为整个程序，故不能在类中内联并定义函数体，否则便会成为局部(即static)的main函数。

第9章

**派生类无虚基类的情况下：（课本P207）**

若派生类的第一个基类建立了虚函数入口地址表（VFT），则派生类就共用该表首址所占用的存储单元；

若派生类的第一个基类没有定义虚函数，派生类就在建立完所有基类的存储空间之后，根据派生类中是否定义了新的虚函数，确定是否为VFT表首址分配一个存储单元，然后为新定义的数据成员建立存储空间。

静态数据成员不包括在内。

**派生类有虚基类的情况下，虚基类的存储空间建于派生类的尾部，且按虚基类的构造顺序建立：**

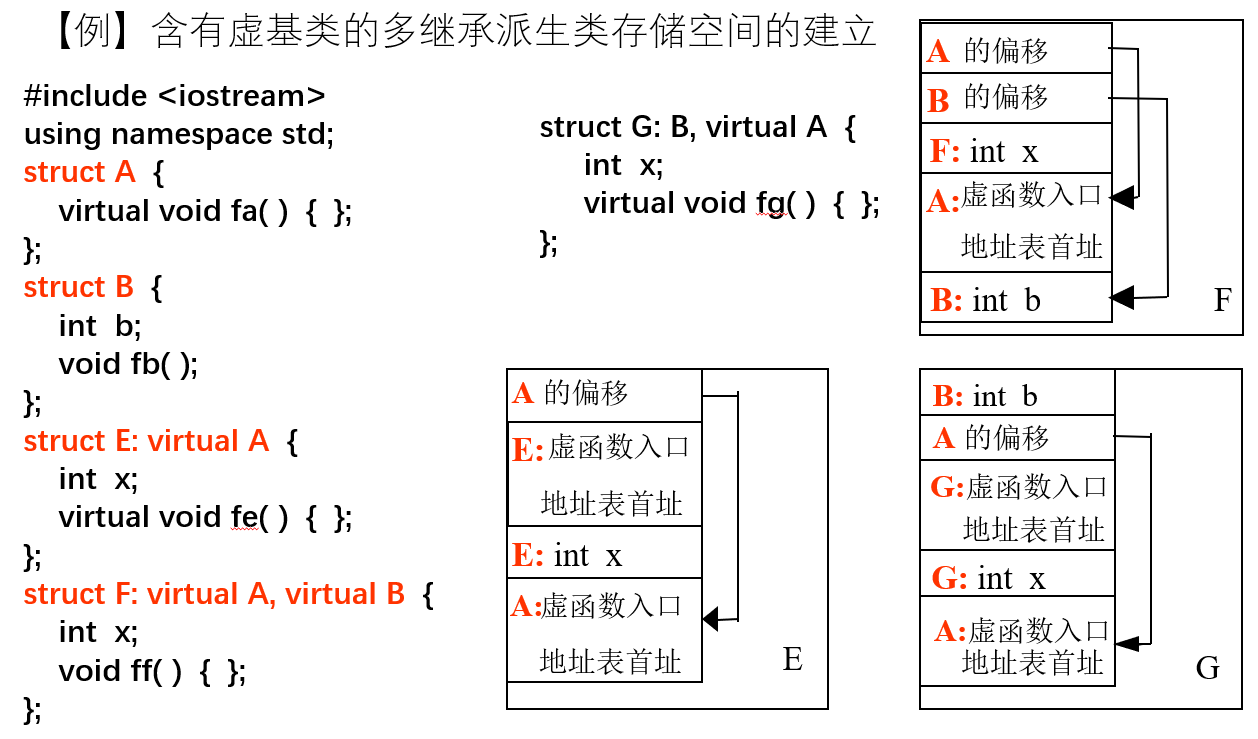
①派生类依次处理每个直接基类或虚基类，如果为直接基类，则为其建立存储空间，如果为直接虚基类则建立一个到虚基类的偏移。

②如果派生类继承的第一个类为非虚基类，且该基类定义了虚函数地址表，则派生类就共享该表首址占用的存储单元。对于其他任何情形，派生类在处理完所有基类或虚基类后，根据派生类是否新定义了虚函数，确定是否为该表首址分配存储单元。

③派生类依次处理自定义的数据成员，为每个数据成员建立相应的存储空间。

④派生类根据虚基类偏移的建立顺序，依次为虚基类建立存储空间，同名虚基类仅在派生类存储空间内建立一次。

⑤如果直接基类和虚基类又是派生类，则在派生类的存储空间内重复①-⑤。如果数据成员又为派生类类型，则在数据成员的存储空间内重复①-⑤。



第13章

**可使用“template 类型 模板名<类型实参>”, 显式生成匿名的实例变量。**

特别注意引用变量实例化，和本体类型变量实例化，二者的存储空间不一样（地址也就不一样）。

