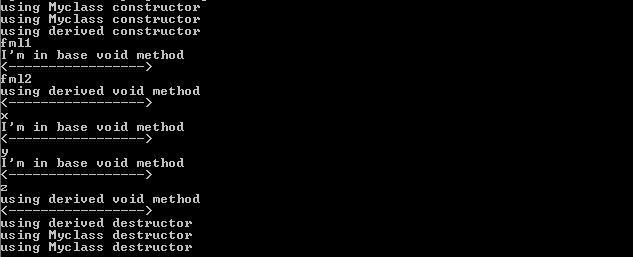
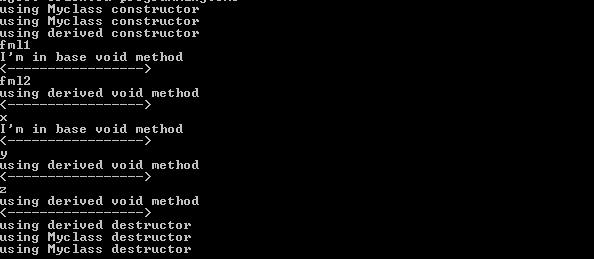
1. Jing Lu, Peiyun(Seed) Zeng

2. Our base class is MyClass and derived class is called Derived. When construct a MyClass instance, it prints “using MyClass constructor” and “using MyClass destructor” when destructed. This is direct result of base class construction and destruction. In the case of Derived class, it prints “using MyClass constructor” and then “using Derived constructor” when constructing an instance of Derived class instance. This is because it first uses constructor from base class and then prints out a message in its own constructor. When destructing the instance, it prints “using Derived destructor” and then “using MyClass destructor”, which implies that it first calls its own destructor and then uses the destructor from base class to clean it up.

3. Output without using virtual keyword:



output using virtual keyword



4. The virtual vs. non-virtual method calls using the pointers acts like references to pointers except we have to dereference or using -> instead of . Why? well, reference is just an auto-pointer that automatically dereference itself.