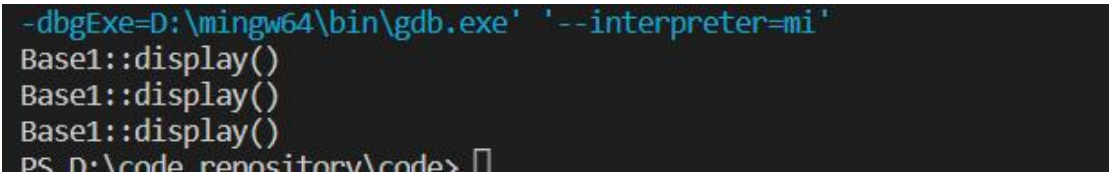
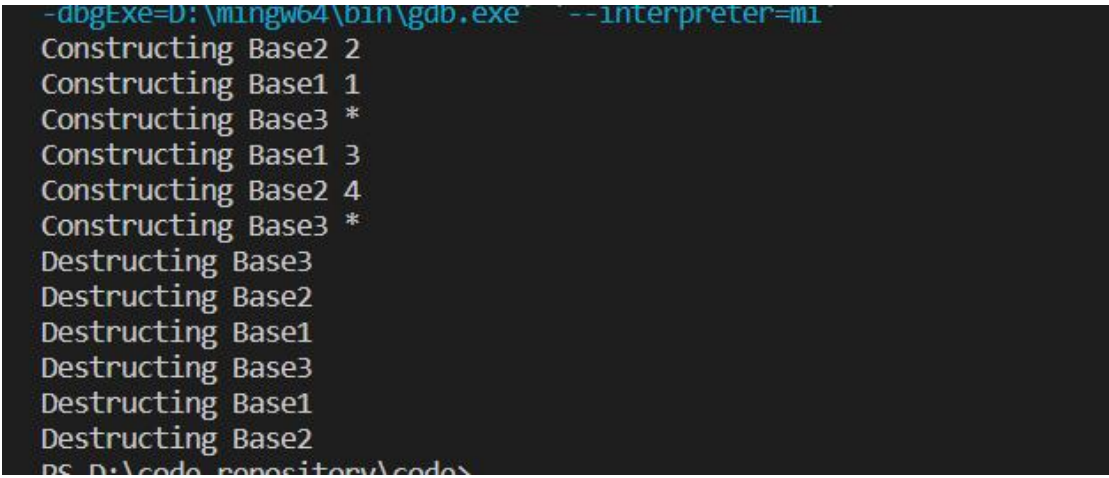


计算机学院_高级语言程序设计_课程实验报告

实验题目：实验九、 多继承，继承中的类型兼容、构造、析构函数执行顺序及访问控制		学号：202200130048
日期：2023. 4. 14	班级： 6	姓名： 陈静雯
Email：1205037094@qq. com		
<p>实验步骤与内容：</p> <ol style="list-style-type: none">1. 练习 7-3. cpp，继承中的类型兼容，截图并解释运行结果。2. 练习例 7-5. cpp， 截图解释运行结果3. 探索继承中的同名隐藏问题。4. 练习第 7_8. cpp，将 main 函数改为如下，解释运行结果。		
<p>结论分析与体会：</p> <ol style="list-style-type: none">1. 2. 		

解释：派生类构造时，先按照继承顺序构造基类，在按照派生类中声明顺序初始化数据成员，析构顺序与构造顺序完全相反

```
7_5.cpp C:\Users\c\j\w\Downloads\7_4
no matching function for call to 'Derived::Derived()' gcc [行 37, 列 13]
```

解释：因为没有对应的构造函数，Derived obj1;声明时没有传对应的参数

3.

```
7_6.cpp C:\Users\c\j\w\Downloads\7_4
no matching function for call to 'Derived::fun(int)' gcc [行 38, 列 12]
no matching function for call to 'Derived::fun(int, int)' gcc [行 39, 列 14]
```

解释：报错，子类会隐藏父类的形参不同的同名函数，子类从父类继承的所有同名的重载函数都会被隐藏

4. (1)

```
-dbgExe=D:\mingw64\bin\gdb.exe' '--interpreter=mi'
Member of Base0
Member of Base0
2 3
PS D:\code_repository\codes>
```

解释：有两个 var0，base1 和 base2 都继承于 base0，derived 继承于 base1 和 base2，构造时会初始化两个 base0 基类，即有两个 var0，分别是 base1 里继承到的和 base2 里继承到的

```
7_7.cpp C:\Users\c\j\w\Downloads\7_4
request for member 'var0' is ambiguous gcc [行 34, 列 13]
'Base0' is an ambiguous base of 'Derived' gcc [行 34, 列 34]
```

分析：报错，因为 derived 的直接基类共同继承于 base0，即 d 中有两个 base0 基类，两个 var0 副本，d.var0 和 d.Base0::var0 会产生二义性，编译器报错

4. (2)

```
-Error-qv5yv401.kal' --pid=Microsoft-MIEngine-Pid-kkvndqgw.n
-dbgExe=D:\mingw64\bin\gdb.exe' '--interpreter=mi'
1 1 1
Member of Base0
2 2 2
3 3 3
PS D:\code_repository\codes>
```

分析：base0 是虚基类，所以对于 base1，base2 来说只有一个 base0 基类，同

名数据成员在内存中只有一个副本，同名函数成员只有一个映射，所以不存在二义性，d 可以直接访问虚基类的数据成员和函数成员，不管是 `derived`，`base1` 还是 `base2`，他们的 `var0` 都是同一个，所以 `d.Base1::var0=1`；`d.Base2::var0=3`；改的是同一个 `var0`，最后输出的 `var0` 是 3