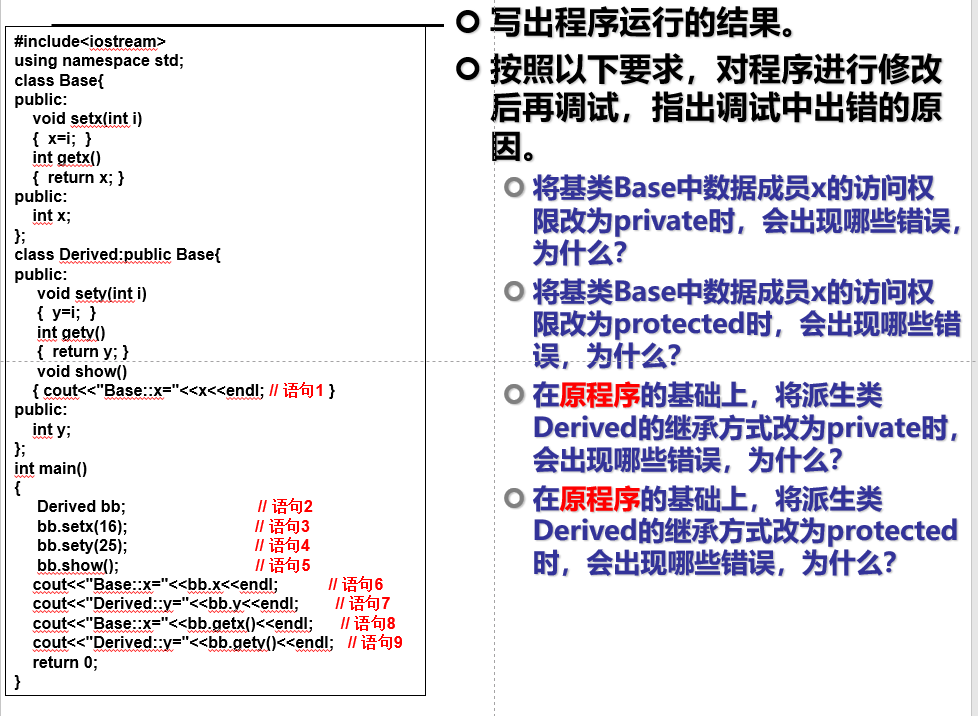
**第五次试验报告**

孙浩然 202031906076

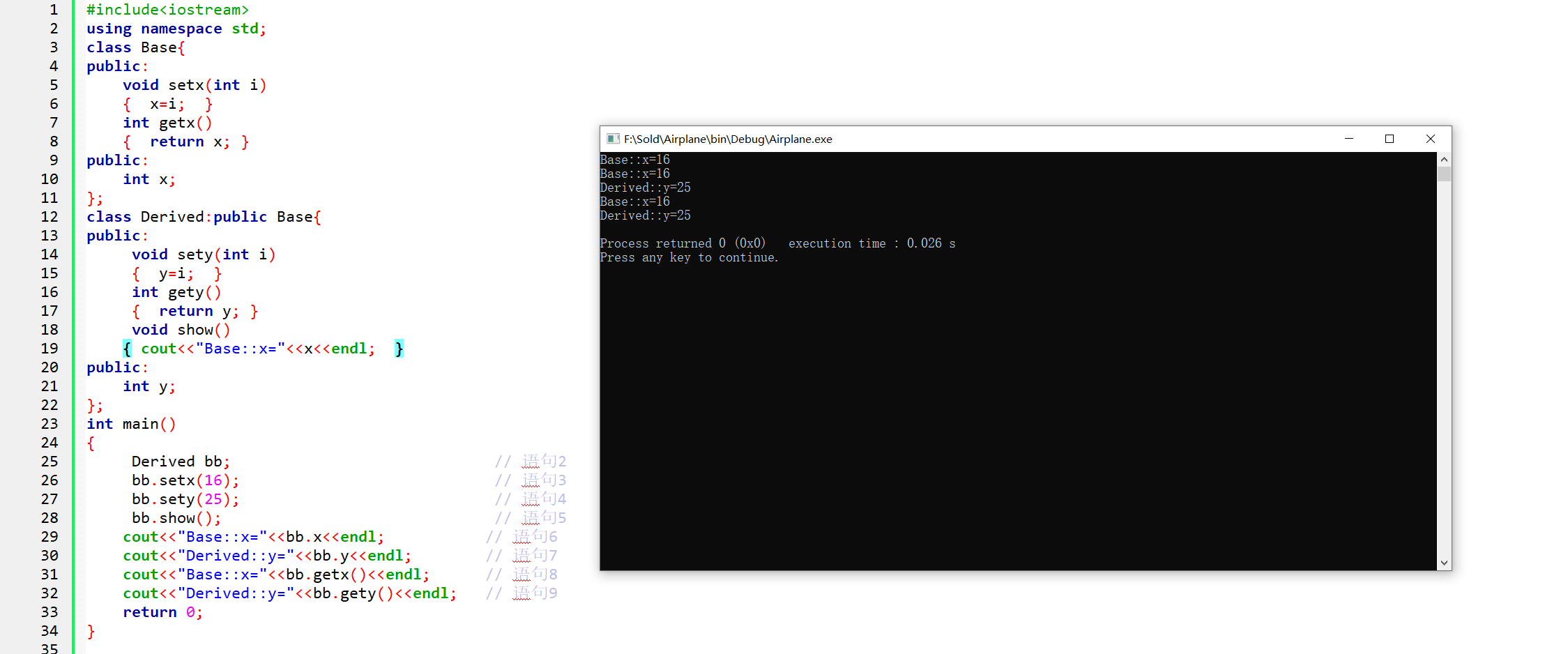
Copyright ©2021-2099 HaoranSun. All rights reserved

**一、实验要求**



**二、心得体会**

第五个试验的完成结果如下：



本次的实验是派生类与继承。

在此先回答所提出的问题

1. 改为private 的话派生类将无法访问成员x，因为这里采用的是公有继承，而private成员是属于基类自己的。
2. 同样的问题，如果改为protected 的话将无法方位base中的x
3. 当采用private继承的时候，派生类将无法访问任何基类的成员，基类的成员对于派生类来说都是private的。
4. 如果采用protected 的话，基类public成员，protected成员，private成员的访问属性在派生类中分别变成：protected, protected, private。

本次实验通过编译器了解了三种继承的不同之处，通过实际的调试让我对这一块的知识有了更加深刻的认识。

**三、实验代码与程序结果**

#include<iostream>

using namespace std;

class Base{

public:

void setx(int i)

{ x=i; }

int getx()

{ return x; }

public:

int x;

};

class Derived:public Base{

public:

void sety(int i)

{ y=i; }

int gety()

{ return y; }

void show()

{ cout<<"Base::x="<<x<<endl; }

public:

int y;

};

int main()

{

Derived bb; // 语句2

bb.setx(16); // 语句3

bb.sety(25); // 语句4

bb.show(); // 语句5

cout<<"Base::x="<<bb.x<<endl; // 语句6

cout<<"Derived::y="<<bb.y<<endl; // 语句7

cout<<"Base::x="<<bb.getx()<<endl; // 语句8

cout<<"Derived::y="<<bb.gety()<<endl; // 语句9

return 0;

}

Copyright ©2021-2099 HaoranSun. All rights reserved