**一程序代码**

#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;

}// 语句1

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

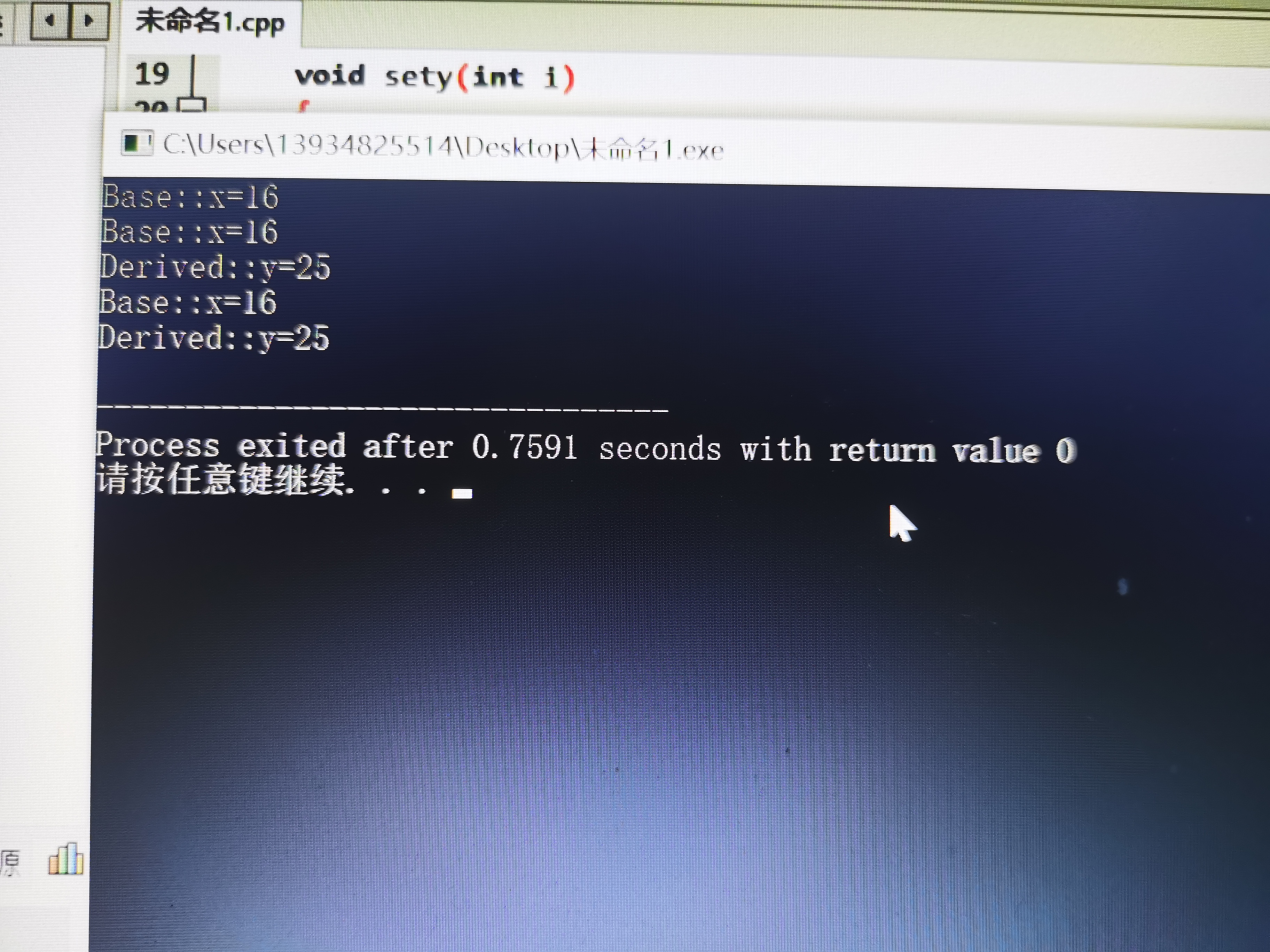
cout << "Base::x=" << bb.getx() << endl; // 语句8

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

return 0;

}

**二结果**



**三心得体会**

**1.当类的继承方式为私有继承时，基类的公有成员和保护成员被继承为派生类的私有成员，派生类的成员函数可以直接访问它，但是在类外部通过派生类的对象则无法访问**

**2.当类的继承方式为公有继承时，基类的共有成员被继承到派生类中仍作为派生类的共有成员，派生类的成员可以直接访问他们，在类的外部，可以通过类的对象直接访问他们**

**3.当类的继承方式为保护继承时，基类的共有成员和保护成员被继承到派生类中都作为派生类中都作为派生类的保护成员，派生类的其他成员可以直接访问他们，在类的外部，不能通过派生类的对象来访问他们**

Copyright ©2021-2099 lvlei. All rights reserved