**派生类与继承**

**一、实验程序**

#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

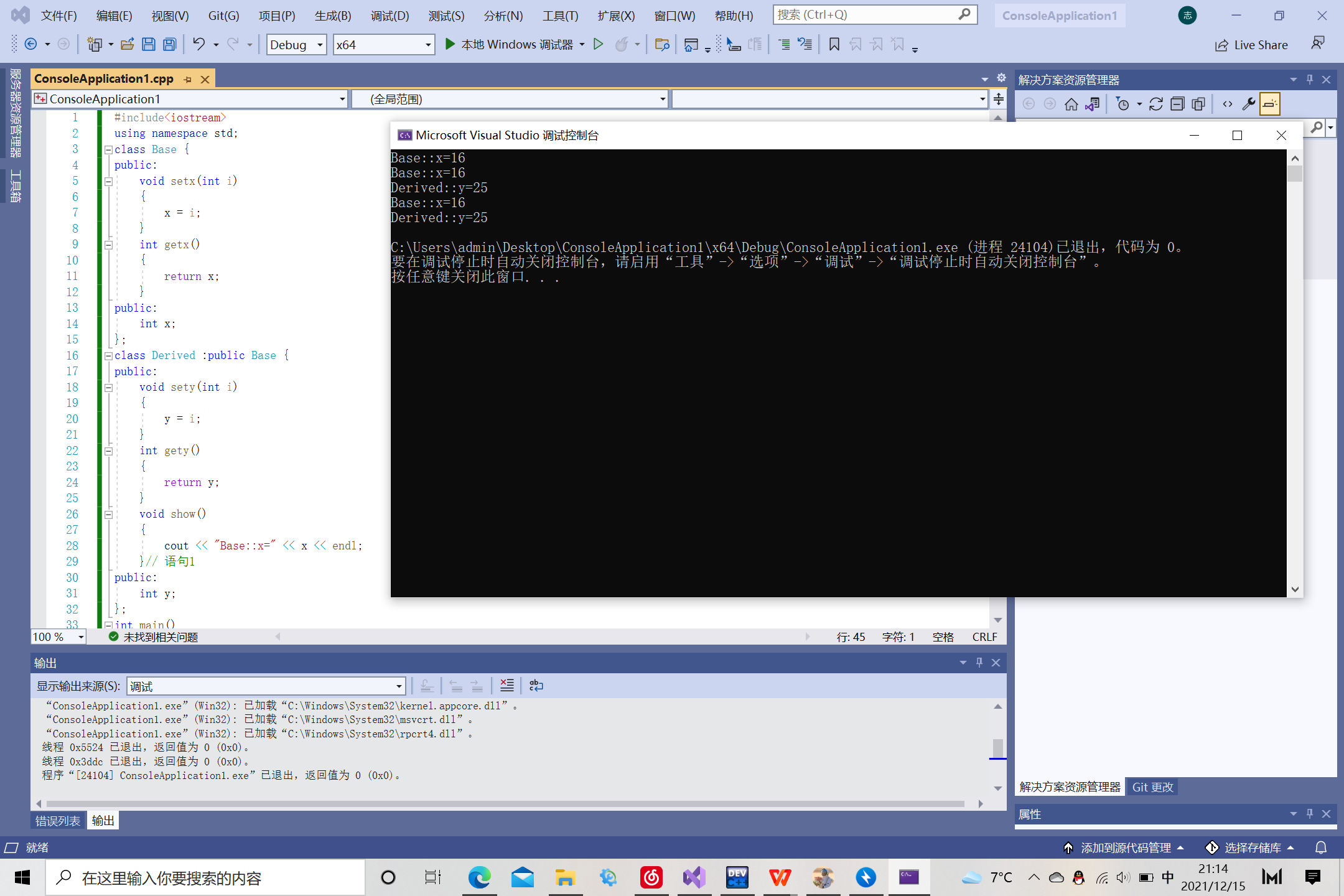
cout << "Base::x=" << bb.getx() << endl; // 语句8

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

return 0;

}

**二、实验结果**



**三、心得体会**

多态性即同一个对象（事物），在不同时刻体现出来的不同状态。父类或接口定义的引用变量可以指向子类或具体实现类的实例对象。

好处：提高了程序的扩展性、灵活性、简化性； 弊端：当父类引用指向子类对象时，虽提高了扩展性，但只能访问父类中具备的方法，不可访问子类中的方法；即访问的局限性。

前提：a:实现或继承关系；

b:方法重写（复写父类方法）；

c:父类引用指向子类对象；

Copyright ©2021-2099 电自2003班 陈阳202030310072. All rights reserved