**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;

bb.setx(16);

bb.sety(25);

bb.show();

cout << "Base::x=" << bb.x << endl;

cout << "Derived::y=" << bb.y << endl;

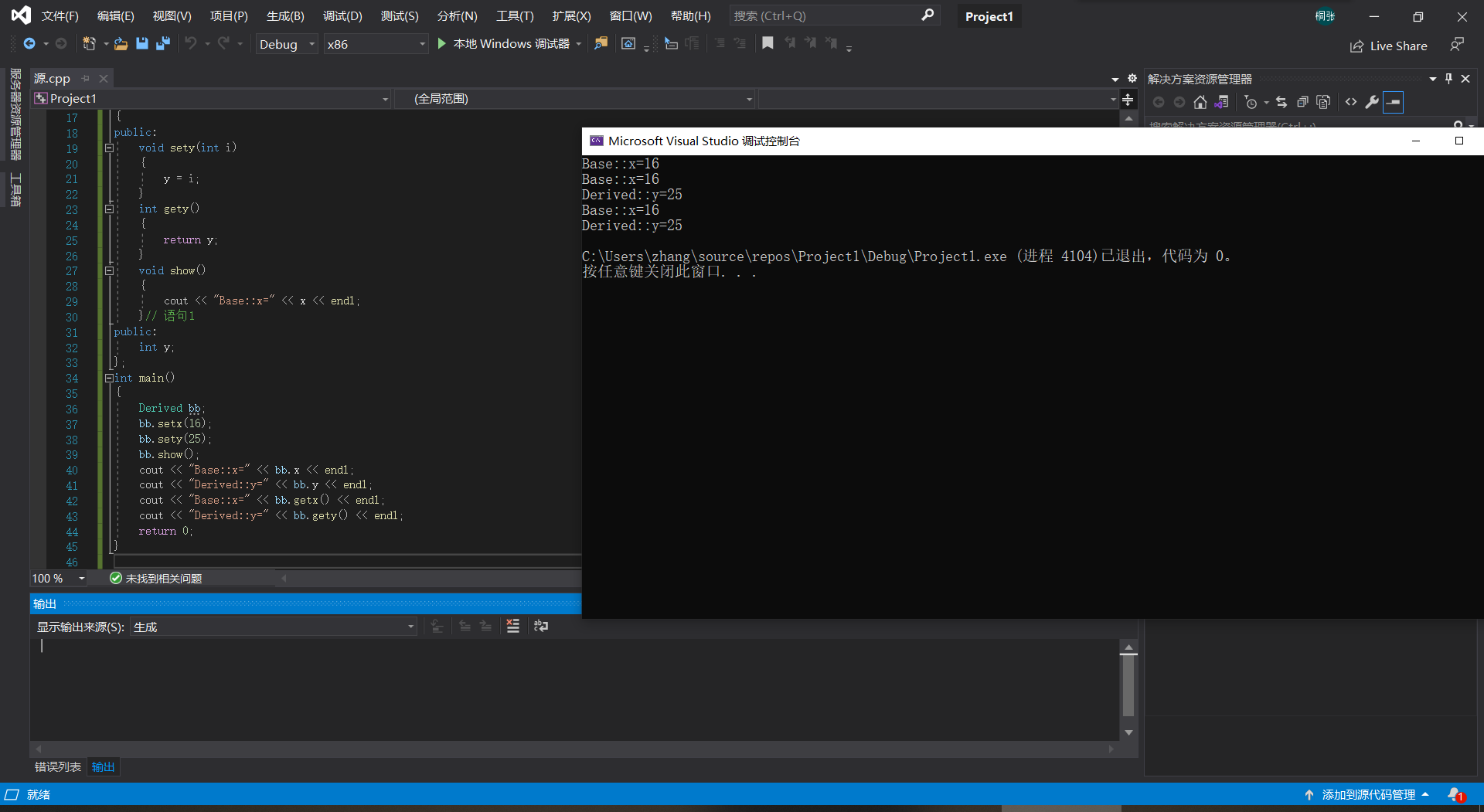
cout << "Base::x=" << bb.getx() << endl;

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

return 0;

}

**二、实验结果**



**三、程序心得**

1. 必须熟练掌握每一种继承方式，在代码的编写过程中做到灵活使用，才能真正实现代码的复用，提高编程效率。
2. 在编写代码的过程中，应当考虑清楚哪些成员函数是要被继承的，以免需要复用函数时出现问题。

Copyright ©2021-2099 TongmingZhang. All rights reserved