第五次实验报告

#include<iostream>

#include<cmath>

#include<string>

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;

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;

}

三种继承方式都无法访问基类的私有成员，（内部和对象两种方式都不行），但都能间接访问基类的公有成员以及保护成员；公有继承还能通过对象直接访问基类中的公有成员，这是其他两种方式无法实现的；私有继承和保护继承都无法通过对象直接访问基类的任意成员。

