

**计算机与信息 学院实验报告**

|  |  |  |
| --- | --- | --- |
| 实验课程： | 面向对象程序设计实验 | |
| 实验编号： | 实验五 | |
| 实验名称： | 子类与继承 | |
| 实验人员： | 学号 | 22111302041 |
| 姓名 | 高蝶 |
| 班级 | 22计算机科学与技术 |
| 指导教师： | 陶涛 | |
| 实验室： | 2号楼202 | |
| 实验日期： | 2023.3.21 | |

# 实验目的

1. 掌握子类的定义；

2. 掌握子类中方法重写；

3. 掌握super关键字，利用super解决子类重写父类成员的问题。

# 实验要求

1. 所编写的程序应符合类的封装、继承要求，符合面向对象程序编写规范；

2. 能利用继承性进行复杂程序的设计和实现；

3. 对问题进行充分的需求分析，设计正确、全面的测试数据，设计类的定义框架及UML图后，再进行编程实现；

4. 所设计的测试数据全部通过后，再进行程序提交，提高程序的通过率。

# 实验内容

**1.验证型实验**

**1.1 子类的定义与使用**

仔细读下面的JAVA语言源程序，自己给出程序的运行结果，再与上机运行程序所得的结果作比较，如果有错，务必弄清楚出错原因。

class Person{

String name;

String sex;

int age;

public Person(String n,String s,int a){

name = n;

sex = s;

age = a;

}

void show(){

System.out.println("name: "+name);

System.out.println("sex: "+sex);

System.out.println("age: "+age);

}

}

class Student extends Person{

String sID;

public Student(String n,String s,int a,String sid){

super(n,s,a);

sID = sid;

}

void showID(){

System.out.println("sID: "+sID);

}

}

public class X {

public static void main(String[] args) {

Student s1=new Student("Zhangsan","Male",20,"102A");

Student s2=new Student("Lisi","Female",18,"108S");

s1.show();

s1.showID();

s2.show();

s2.showID();

}

}

**1.2方法重写**

仔细读下面的JAVA语言源程序，自己给出程序的运行结果，再与上机运行程序所得的结果作比较，如果有错，务必弄清楚出错原因。

//CastDemo1.java

class Employee{

public String toString() {

return "I am an Employee";

}

}

class Manager extends Employee{

public String toString(){

return "I'm a manager.";

}

}

public class CastDemo1{

public static void main(String args[]){

Employee stuff;

Manager boss=new Manager();

stuff=new Employee();

System.out.println(stuff);

stuff=boss;

System.out.println(boss);

System.out.println(stuff);

}

}

**2.应用型实验**

**2.1矩形计算**

要求编写一个Java应用程序：

（1）编写一个矩形类Rect，包含：

两个protected属性：矩形的宽width；矩形的高height。

两个构造方法：

一个带有两个参数的构造方法，用于将width和height属性初化；

一个不带参数的构造方法，将矩形初始化为宽和高都为10。

两个方法：

求矩形面积的方法area()

求矩形周长的方法perimeter()

（2）通过继承Rect类编写一个具有确定位置的矩形类PlainRect，其确定位置用矩形的左上角坐标来标识，包含：

添加两个属性：矩形左上角坐标startX和startY。

两个构造方法：

带4个参数的构造方法，用于对startX、startY、width和height属性初始化；

不带参数的构造方法，将矩形初始化为左上角坐标、长和宽都为0的矩形；

添加一个方法：

判断某个点是否在矩形内部的方法isInside(double x,double y)。如在矩形内或点在边上，返回true, 否则，返回false。

（3）编写Main类的测试程序

创建一个左上角坐标为（x，y），宽为w，高为h的矩形对象；计算并打印输出矩形的面积和周长；判断点(m，n)是否在矩形内，并打印输出相关信息（In或Out）。

输入：左上角坐标，宽和高，点的坐标

输出：面积、周长与是否在矩形内

样例输入：

8.72 8.36

19.25 9.57

11.04 17.02

样例输出：

184.2225 57.64 Out

**2.2 点面体**

编写程序实现如下功能：

（1）设计一个表示二维平面上点的类Point，包含有表示坐标位置的protected 类型的成员变量x 和y，获取和设置x 和y 值的public 方法。

（2）设计一个表示二维平面上圆的类Circle，它继承自类Point，还包含有表示圆半径的protected 类型的成员变量r、获取和设置r 值的public 方法、计算圆面积的public 方法。

（3）设计一个表示圆柱体的类Cylinder，它继承自类Circle，还包含有表示圆柱体高的protected 类型的成员变量h、获取和设置h 值的public 方法、计算圆柱体体积的public方法。

（4）在主类中设计菜单显示，表示输入圆或圆柱体，1表示圆，2表示圆柱体，0表示运行结束。输入的是圆时，需输入圆心位置、半径；输入的是圆柱体时，需输入底部圆心位置、半径与高。

说明：类中的其他方法可自行定义

输入：n个圆对象与m个圆柱体对象，成员变量类型为double型。

输出：圆的圆心、半径与面积值，圆柱体的底部圆的信息、高与体积。圆心、半径、高等值输出时保留1位小数，面积输出时保留2位小数，体积输出时保留4位小数。圆周率π使用Math.PI。

样例输入：

1

2 3 5

2

1 2 4 3

0

样例输出：

please input(0-2):

(2.0,3.0);5.0;78.54

please input(0-2):

(1.0,2.0);4.0;50.27;3.0;150.7964

please input(0-2):

**3. 设计型实验**

**3.1 高考成绩排序**

高考生主要分为文科生与理科生，文科生的考试科目为：语文、外语、数学和文综，理科生的考试科目为：语文、外语、数学和理综。但是参加高考考试的还有一部分考生称为艺体生，他们除了参加文化课考试外还要参加专业考试。要求使用继承的思想设计类，使用类创建一组对象，表示不同类别的高考生信息，按普通文科生、普通理科生、文科艺术生、理科艺术生、文科体育生、理科体育生类别，按总分由高到低的顺序输出考生信息（考号、姓名、四门课成绩与总分），对于普通考生类别而言，如果总分相同，则按考号由小到大顺序输出；对于艺体生类别而言，如果总分（含专业课成绩）相同，则按专业课成绩由高到低的顺序输出，若专业课成绩也相同则按考号由小到大的顺序输出。

要求使用类的继承完成程序设计。

输入：普通文理生人数，普通文理生信息，艺体生人数，艺体生信息（普通文理生文理分类要求：1-文，2-理；艺体生先有文理分类，再有艺术、体育分类：1-艺术，2-体育）

输出：排序后的考生信息包括：学号,姓名,各科分数,总分,类别

样例输入：

8

0107 xpc 1 131 103 128 276

0120 pfn 2 92 97 144 278

0477 qok 2 145 113 148 267

0006 rel 1 123 126 144 273

0972 orz 2 106 104 114 280

0380 yor 1 128 132 91 230

0470 wpl 2 128 123 94 226

0203 lmw 2 102 111 145 272

8

1695 sxc 1 2 124 104 124 245 92

1463 yzz 2 1 112 100 140 279 88

1357 qbg 1 1 137 104 117 276 100

1168 fio 2 2 94 145 138 271 81

1085 ffz 2 1 147 108 120 220 93

1784 tkm 1 1 147 120 142 278 95

1595 cdd 2 2 139 150 120 221 91

1886 hzf 1 1 148 139 146 218 87

样例输出：

0006,rel,123,126,144,273,666,1

0107,xpc,131,103,128,276,638,1

0380,yor,128,132,91,230,581,1

0477,qok,145,113,148,267,673,2

0203,lmw,102,111,145,272,630,2

0120,pfn,92,97,144,278,611,2

0972,orz,106,104,114,280,604,2

0470,wpl,128,123,94,226,571,2

1784,tkm,147,120,142,278,95,782,11

1886,hzf,148,139,146,218,87,738,11

1357,qbg,137,104,117,276,100,734,11

1463,yzz,112,100,140,279,88,719,12

1085,ffz,147,108,120,220,93,688,12

1695,sxc,124,104,124,245,92,689,21

1168,fio,94,145,138,271,81,729,22

1595,cdd,139,150,120,221,91,721,22

**3.2 银行利息计算**

国家规定了银行存款的定期基准利率，分别为：1年期的每年利率为1.5%，2年期的每年利率为2.1%，3年期与5年期的每年利率均为2.75%，用户根据自己的情况选择不同的存款年限，满期取钱时取得的本息值计算方法为：本金+本金\*相应的利率\*年数。银行提供自动转存服务，即到期不取款仍按原存期转存，这时以前的利息记入本金，例如：2年期存款1万元，4年到期取款时本息值计算方法为：10000\*(1+0.021\*2)^2。

国有银行的定期利率分别是：1年期的每年利率为1.75%，2年期的每年利率为2.25%，3年期与5年期的每年利率均为2.75%，用户取定期存款时，首先得明确是几年期的，存款的时限（以年为单位），例如，3年期的，存款的时限为7.35，7.35理解为7年350天，银行在计算利率时，不足存款期限年份值的按活期利率0.001%（日利率）计算，活期利息的计算方法为：本金\*日利率\*存款天数，其中，本金是指按活期计算前的本息值，1年按365天计算，则存7.35年的3年期在取款时1.35年的时间按活期利率计算。

民营银行的定期利率分别是：1年期的每年利率为1.95%，2年期的每年利率为2.41%，3年期的每年利率为2.75%与5年期的每年利率为3.00%，用户取定期存款时，首先得明确是几年期的，存款的时限（以年为单位），例如，3年期的，存款的时限为7.35，7.35理解为7年350天，银行在计算利率时，不足存款期限年份值的不满一年的（上例中即为350）按活期利率0.0012%（日利率）计算，活期利息的计算方法为：本金\*日利率\*存款天数，其中，本金是指按活期计算前的本息值。不足存款期限的年份值超过1年以上的年数值（上例中即为1）按1年期定期存款利率计息，（不计复利，本金是按一年期计算前的本息值），剩下的天数按活期利率0.0012%（日利率）计息（本金是指按活期计算前的本息值）。

按照类的继承思想，设计三个银行类，再设计主类，依据输入的银行类型（1-国有银行，2-民营银行）、存款数、几年期（数字1、2、3、5中的一个）及其存款时限，输出相应的本息值（保留1位小数）。

输入：银行类型（1-国有银行，2-民营银行）、存款数、几年期（数字1、2、3、5中的一个）及其存款时限

输出：本息值（保留1位小数）

样例输入：1 10000 2 5.2

样例输出：10981.9

# 程序清单

2.应用型实验

（1）

**测试数据：**

第1组：8.72 8.36

19.25 9.57

11.04 17.02

第2组：8.36 8.72

2.00 4.00

5.90 6.34

第3组：4.89 6.45

7.45 8.67

6.87 4.45

**程序源码：package** prj\_juxingjisuan;

**import** java.util.Scanner;

**public** **class** Main {

**public** **static** **void** main(String[] args) {

Scanner reader=**new** Scanner(System.***in***);

**double** x,y,w,h,m,n;

x=reader.nextDouble();

y=reader.nextDouble();

w=reader.nextDouble();

h=reader.nextDouble();

m=reader.nextDouble();

n=reader.nextDouble();

PlainRect p =**new** PlainRect(w,h,x,y);

p.getArea();

p.getPerimeter();

**if**(p.isInside(m,n)) System.***out***.printf("In");

**else** System.***out***.printf("Out");

}

}

**class** Rect{

**protected** **double** width,height;

Rect (**double** width,**double** height){

**this**.width=width;

**this**.height=height;

}

Rect(){

width=10;

height=10;

}

**public** **void** getArea() {

System.***out***.printf("%.4f ",width\*height);

}

**public** **void** getPerimeter() {

System.***out***.printf("%.2f ",(width+height)\*2);

}

}

**class** PlainRect **extends** Rect{

**double** startX,startY;

PlainRect(**double** width,**double** height,**double** startX,**double** startY){

**super**(width,height);

**this**.startX=startX;

**this**.startY=startY;

}

PlainRect(){

**super**(0,0);

startX=0;

startY=0;

}

**boolean** isInside(**double** x,**double** y) {

**double** endX=startX+width,endY=startY-height;

**if**(x>=startX&&x<=endX&&y>=endY&&y<=startY) **return** **true**;

**else** **return** **false**;

}

}

（2）

**测试数据：**

第1组：1

2 3 5

2

1 2 4 3

0

第2组：2

1 2 3 4

1

2 3 4

0

第3组：1

1 3 4 5

1

2 3 4 5

0

**程序源码：package** prj\_dianmianti;

**import** java.util.Scanner;

**public** **class** Main {

**public** **static** **void** main(String[] args) {

Scanner reader=**new** Scanner(System.***in***);

**int** x;

System.***out***.println("please input(0-2):");

x=reader.nextInt();

**while**(x!=0) {

**double** xx,yy,r,h;

xx=reader.nextDouble();

yy=reader.nextDouble();

r=reader.nextDouble();

**if**(x==1) {

Circle p=**new** Circle(xx,yy,r);

p.print();

}

**else** **if**(x==2){

h=reader.nextDouble();

Cylinder q=**new** Cylinder(xx,yy,r,h);

q.print();

}

System.***out***.println("please input(0-2):");

x=reader.nextInt();

}

}

}

**class** Point{

**protected** **double** x,y;

Point(**double** x,**double** y){

**this**.x=x;

**this**.y=y;

}

**public** **void** setx(**double** x) {

**this**.x=x;

}

**public** **void** sety() {

**this**.y=y;

}

**public** **double** getx() {

**return** x;

}

**public** **double** gety() {

**return** y;

}

}

**class** Circle **extends** Point{

**protected** **double** r;

Circle(**double** x,**double** y,**double** r){

**super**(x,y);

**this**.r=r;

}

**public** **void** setr(**double** r) {

**this**.r=r;

}

**public** **double** getr() {

**return** r;

}

**public** **double** getArea() {

**return** Math.***PI***\*r\*r;

}

**void** print() {

System.***out***.printf("(%.1f,%.1f);%.1f;%.2f\n",x,y,r,getArea());

}

}

**class** Cylinder **extends** Circle{

**protected** **double** h;

Cylinder(**double** x,**double** y,**double** r,**double** h){

**super**(x,y,r);

**this**.h=h;

}

**public** **void** seth(**double** h) {

**this**.h=h;

}

**public** **double** geth() {

**return** h;

}

**public** **double** getV() {

**return** getArea()\*h;

}

**void** print() {

System.***out***.printf("(%.1f,%.1f);%.1f;%.2f;%.1f;%.4f\n",x,y,r,getArea(),h,getV());

}

}

3.设计型实验

（1）

**测试数据：**

第1组：8

0107 xpc 1 131 103 128 276

0120 pfn 2 92 97 144 278

0477 qok 2 145 113 148 267

0006 rel 1 123 126 144 273

0972 orz 2 106 104 114 280

0380 yor 1 128 132 91 230

0470 wpl 2 128 123 94 226

0203 lmw 2 102 111 145 272

8

1695 sxc 1 2 124 104 124 245 92

1463 yzz 2 1 112 100 140 279 88

1357 qbg 1 1 137 104 117 276 100

1168 fio 2 2 94 145 138 271 81

1085 ffz 2 1 147 108 120 220 93

1784 tkm 1 1 147 120 142 278 95

1595 cdd 2 2 139 150 120 221 91

1886 hzf 1 1 148 139 146 218 87

第2组：3

0107 xpc 1 131 103 128 276

0120 pfn 2 92 97 144 278

0477 qok 2 145 113 148 267

2

1357 qbg 1 1 137 104 117 276 100

1168 fio 2 2 94 145 138 271 81

第3组：2

0470 wpl 2 128 123 94 226

0203 lmw 2 102 111 145 272

3

1168 fio 2 2 94 145 138 271 81

1085 ffz 2 1 147 108 120 220 93

1784 tkm 1 1 147 120 142 278 95

**程序源码：package** prj\_gaokaoscore;

**import** java.util.Scanner;

**public** **class** Main {

**public** **static** **void** main(String[] args) {

Scanner reader=**new** Scanner(System.***in***);

**int** n,m,i;

String no,name;

**int** index1,index2,csco,esco,msco,culsco,majsco;

n=reader.nextInt();

wenliStudent[] stu=**new** wenliStudent[n+10];

**for**(i=0;i<n;i++) {

no=reader.next();

name=reader.next();

index1=reader.nextInt();

csco=reader.nextInt();

esco=reader.nextInt();

msco=reader.nextInt();

culsco=reader.nextInt();

stu[i]=**new** wenliStudent(no,name,csco,esco,msco,index1,culsco);

stu[i].setsumsco();

}

m=reader.nextInt();

wenliyitiStudent[] sstu=**new** wenliyitiStudent[m+10];

**for**(i=0;i<m;i++) {

no=reader.next();

name=reader.next();

index1=reader.nextInt();

index2=reader.nextInt();

csco=reader.nextInt();

esco=reader.nextInt();

msco=reader.nextInt();

culsco=reader.nextInt();

majsco=reader.nextInt();

sstu[i]=**new** wenliyitiStudent(no,name,csco,esco,msco,index1,culsco,index2,majsco);

sstu[i].setsumsco();

}

wenliStudent.*sort*(stu,n);

**for**(i=0;i<n;i++) stu[i].print();

wenliyitiStudent.*sort*(sstu,m);

**for**(i=0;i<m;i++) sstu[i].print();

}

}

**class** wenliStudent{

**protected** String no,name;

**protected** **int** index1,chinesesco,englishsco,mathsco,culturesco,sumsco;

wenliStudent(){};

wenliStudent(String no,String name,**int** chinesesco,**int** englishsco,**int** mathsco ,**int** index1,**int** culturesco){

**this**.no=no;

**this**.name=name;

**this**.chinesesco=chinesesco;

**this**.englishsco=englishsco;

**this**.mathsco=mathsco;

**this**.index1=index1;

**this**.culturesco=culturesco;

}

**void** setsumsco() {

sumsco=chinesesco+englishsco+mathsco+culturesco;

}

**static** **void** sort(wenliStudent[] stu,**int** n){

**int** i,j;

wenliStudent t=**new** wenliStudent();

**for**(i=1;i<=n-1;i++) {

**for**(j=1;j<=n-i;j++) {

**if**(stu[j-1].index1>stu[j].index1||stu[j-1].index1==stu[j].index1&&stu[j-1].sumsco<stu[j].sumsco||stu[j-1].index1==stu[j].index1&&stu[j-1].sumsco==stu[j].sumsco&&stu[j-1].no.compareTo(stu[j].no)>0) {

t=stu[j-1];

stu[j-1]=stu[j];

stu[j]=t;

}

}

}

}

**void** print() {

System.***out***.printf("%s,%s,%d,%d,%d,%d,%d,%d\n",no,name,chinesesco,englishsco,mathsco ,culturesco,sumsco,index1);

}

}

**class** wenliyitiStudent **extends** wenliStudent{

**protected** **int** majorsco,index2,ssumsco=0,index=0;

wenliyitiStudent(){}

wenliyitiStudent(String no,String name,**int** chinesesco,**int** englishsco,**int** mathsco ,**int** index1,**int** culturesco,**int** index2,**int** majorsco){

**super**(no,name,chinesesco,englishsco,mathsco,index1,culturesco);

**this**.index2=index2;

**this**.majorsco=majorsco;

}

**void** setsumsco() {

ssumsco=chinesesco+englishsco+mathsco+culturesco+majorsco;

index=index2\*10+index1;

}

**static** **void** sort(wenliyitiStudent[] sstu,**int** m){

**int** i,j;

wenliyitiStudent t=**new** wenliyitiStudent();

**for**(i=1;i<=m-1;i++) {

**for**(j=1;j<=m-i;j++) {

**if**(sstu[j-1].index>sstu[j].index||sstu[j-1].index==sstu[j].index&&sstu[j-1].ssumsco<sstu[j].ssumsco||sstu[j-1].index==sstu[j].index&&sstu[j-1].ssumsco==sstu[j].ssumsco&&sstu[j-1].majorsco<sstu[j].majorsco||sstu[j-1].index==sstu[j].index&&sstu[j-1].ssumsco==sstu[j].ssumsco&&sstu[j-1].majorsco==sstu[j].majorsco&&sstu[j-1].no.compareTo(sstu[j].no)>0) {

t=sstu[j-1];

sstu[j-1]=sstu[j];

sstu[j]=t;

}

}

}

}

**void** print() {

System.***out***.printf("%s,%s,%d,%d,%d,%d,%d,%d,%d%d\n",no,name,chinesesco,englishsco,mathsco ,culturesco,majorsco,ssumsco,index2,index1);

}

}

（2）

**测试数据：**

第1组：1 10000 2 5.2

第2组：2 10000 2 5.2

第3组：1 20000 3 7.4

**程序源码：package** prj\_bankjisuan;

**import** java.util.Scanner;

**public** **class** Main {

**public** **static** **void** main(String[] args) {

Scanner reader=**new** Scanner(System.***in***);

**int** index,ytype;

**double** capital,year;

index=reader.nextInt();

capital=reader.nextDouble();

ytype=reader.nextInt();

year=reader.nextDouble();

**if**(index==1) {

publicBank p=**new** publicBank(capital,ytype,year);

p.print();

}**else** {

minBank q=**new** minBank(capital,ytype,year);

q.print();

}

}

}

**class** Bank{

**int** ytype;

**double** year,capital;

**double** [] interest= {0,0.015,0.021,0.0275,0,0.0275};

Bank(**double** capital,**int** ytype,**double** year){

**this**.capital=capital;

**this**.ytype=ytype;

**this**.year=year;

}

}

**class** publicBank **extends** Bank{

**double** [] interest1={0,0.0175,0.0225,0.0275,0,0.0275};

**double** dayrate=0.00001;

publicBank(**double** capital,**int** ytype,**double** year){

**super**(capital,ytype,year);

}

**void** print() {

**double** sum=capital;

**int** j=(**int**)year/ytype,day=(**int**)year%ytype\*365+(**int**)(1000\*(year-(**int**)year));

**for**(**int** i=1;i<=j;i++) {

sum+=sum\*interest1[ytype]\*ytype;

}

//System.out.println(sum+" "+day);

sum+=day\*dayrate\*sum;

System.***out***.printf("%.1f",sum);

}

}

**class** minBank **extends** Bank{

**double** [] interest2={0,0.0195,0.0241,0.0275,0,0.03};

**double** dayrate=0.000012;

minBank (**double** capital,**int** ytype,**double** year){

**super**(capital,ytype,year);

}

**void** print() {

**double** sum=capital;

**int** j=(**int**)year/ytype,y=(**int**)year%ytype,day=(**int**)(1000\*(year-(**int**)year));

**for**(**int** i=1;i<=j;i++) {

sum+=sum\*interest2[ytype]\*ytype;

}

//System.out.println(sum+" "+day);

sum+=y\*interest2[1]\*sum;

sum+=day\*dayrate\*sum;

System.***out***.printf("%.1f",sum);

}

}

# 实验结果及分析

1.验证型实验

（1）分析本程序的执行过程，给出其运行结果

执行过程：创建两个student子类的对象，调用父类的构造器完成初始化，再借助父类的show方法和自己的方法分别输出信息。

运行结果 name: Zhangsan

sex: Male

age: 20

sID: 102A

name: Lisi

sex: Female

age: 18

sID: 108S

1. 分析本程序的执行过程，给出其运行结果

执行过程：创建子类父类对象，输出父类信息，将子类对象引用复制给父类对象，再分别输出它们的信息。

运行结果：I am an Employee

I'm a manager.

I'm a manager.

2.应用型实验

（1）

各个类、方法的作用：主类：输入调用输出；Rect类定义宽高及获得面积和周长；PlainRect类判断点是否在矩形内。

运行结果：1：184.2225 57.64 Out

2：8.0000 12.00 Out

3：64.5915 32.24 In

（2）

各个类、方法的作用：主类：输入调用输出；Point定义点的坐标；Circle子类定义半径和算出面积；Cylinder子类定义高和计算体积。

运行结果：1：please input(0-2):

(2.0,3.0);5.0;78.54

please input(0-2):

(1.0,2.0);4.0;50.27;3.0;150.7964

please input(0-2):

2：please input(0-2):

(1.0,2.0);3.0;28.27;4.0;113.0973

please input(0-2):

(2.0,3.0);4.0;50.27

please input(0-2):

3：please input(0-2):

(1.0,3.0);4.0;50.27

please input(0-2):

(2.0,3.0);4.0;50.27

please input(0-2):

3.设计型实验

（1）

各个类、方法的作用：主类：输入调用输出；wenliStudent父类完成文理考生信息的定义与排名；wenliyitiStudent 子类在父类基础上补充艺体生的信息及排名方法。

运行结果：1：0006,rel,123,126,144,273,666,1

0107,xpc,131,103,128,276,638,1

0380,yor,128,132,91,230,581,1

0477,qok,145,113,148,267,673,2

0203,lmw,102,111,145,272,630,2

0120,pfn,92,97,144,278,611,2

0972,orz,106,104,114,280,604,2

0470,wpl,128,123,94,226,571,2

1784,tkm,147,120,142,278,95,782,11

1886,hzf,148,139,146,218,87,738,11

1357,qbg,137,104,117,276,100,734,11

1463,yzz,112,100,140,279,88,719,12

1085,ffz,147,108,120,220,93,688,12

1695,sxc,124,104,124,245,92,689,21

1168,fio,94,145,138,271,81,729,22

1595,cdd,139,150,120,221,91,721,22

2：0107,xpc,131,103,128,276,638,1

0477,qok,145,113,148,267,673,2

0120,pfn,92,97,144,278,611,2

1357,qbg,137,104,117,276,100,734,11

1168,fio,94,145,138,271,81,729,22

3：0203,lmw,102,111,145,272,630,2

0470,wpl,128,123,94,226,571,2

1784,tkm,147,120,142,278,95,782,11

1085,ffz,147,108,120,220,93,688,12

1168,fio,94,145,138,271,81,729,22

（2）

各个类、方法的作用：主类完成输入调用；Bank定义共有属性；publicBank定义国有银行的特有相关信息和计算；minBank定义民营银行的特有信息和计算。

运行结果：1：10981.9

2：11228.4

3：23615.4

# 实验小结

[本次实验的反思、收获和体会]

反思：1：高考成绩排序一题中对排序有多重要求，我一开始的逻辑写起来麻烦且重复排序了，要求之间的逻辑关系未理清，后来问了同学才明白过来只需将排序先后需求一一列举出并在一起即可。

2：银行利息计算的题目较长有点难以理解，需要耐下心慢慢看。其中的强制类型转换的一些小细节很容易出错。还有要理清到底是怎么存钱的。

收获：1：弄清了多要求排序的方法。

2：写题过程尝试了类变量get set方法的快捷键感觉很有用。

3：加深了继承的一些写法的理解，如父类成员的类型设定，子类对象的初始化，super的使用等。

体会：这次的题目感觉偏向于继承的使用规范的练习，边听课边自己敲代码做题才能更好理解掌握。感觉继承确实挺好用的，不用重复定义成员，代码不冗余。但要熟练使用还得再学学相关的知识。