**《Java程序设计》实验报告**

**学生姓名： 耿立博 班级： 19计算机3班 学号： 2019207320123**

**实验日期： 2021.5.11 指导教师： 胡继礼、谷宗运、李芳芳**

**实验环境：win7+Jdk1.8**

**一、实验目的**

**掌握** 1. 掌握继承的概念与实现方法；

2. 掌握包和接口的定义和使用方法；

3. 掌握抽象类的定义和使用

4. 掌握异常处理方法;

5. 正确地使用捕获异常和声明抛弃异常的两种异常处理的方法;

**了解 1.**了解JAVA语言实现多继承的途径；

**2.**了解自定义异常类的使用;

**二、实验内容**

**新建java工程test，以下所有类包名为cn.edu.ahtcm.bean**

1.定义一个昆虫类Insect，要求如下

（1）包括属性name，size，color

（2）定义其有参构造方法;

（3）生成成员变量的get和set方法；

（4）定义run方法 void run(Insect a)打印输入谁在跑，接收参数为Insect类型；

（5）在main函数测试该类new一个实例对象i，调用run方法，然后打印输出i的name，size，color

2.定义一个接口IAttack

（1）包含静态的不可改变的字符串常量值ADDRESS字符串，内容为"China";

（2）包含两个方法move（）和attack（）

（3）包含默认方法default void grow()实现内容为打印输出"Insect grows fastly"

3.定义一个Ant类,要求如下

（1）继承Insect，实现接口IAttack

（2）实现move（）方法：打印输出"Ant move slowly"；

（3）实现attack（）方法：打印输出"Ant bite people"；

（4）重写父类的run（）方法，打印输出"Ant walk step by step";

（4）在main函数new一个Ant实例对象，赋值给对象a，

Ant a = new Ant(…)

调用move和attack、run、grow方法

4.定义一个抽象类AbsAnimal，要求如下

（1）包括属性name，weight，color;

（2）定义其有参构造方法;

（3）定义受保护的run方法，功能是打印输出"Animal run fastly";

（4）在main方法里实例化抽象对象a

AbsAnimal a = new AbsAnimal(….) ，想一想为什么会报错？如果报错注释掉该语句。

5.定义一个类Tiger继承抽象类AbsAnimal，要求如下

（1）在main方法中实例化一个Tiger对象

（2）调用run方法，查看输出结果

6.定义一个类TestDivisionException,要求如下

（1）定义静态方法divisionNum，接收输入的两个整数数，返回整除结果，使用throws抛出ArithmeticException异常；

（2）main方法里接收键盘输入两个整数a,b;调用divisionNum方法，结果存入变量c，

（3）如果除数为0，对异常进行捕获；

（4）打印输出错误信息和错误堆栈信息；

（5）最后不管是否有错误都打印输出c的结果，并打印“计算完毕”。

7.定义一个类TestArrayBoundException,要求如下

（1）定义字符串数组s，里面存放{“Apple”,”pear”,”banana”}；

（2）循环打印输出每个元素，对于下标越界进行异常捕获；

（3）最后不管是否有错误都打印输出“打印完毕”。

**三、实验过程（按上述实验内容要求写出源代码及运行结果）**

**1.实验-1**

**【源码】**

**package** cn.edu.ahtcm.bean;

/\*

2019207320123

耿立博

21.5.11

\*/

**public** **class** Insect {

//（1）包括属性name，size，color

String name;

**int** size;

String color;

//（2）定义其有参构造方法;

**public** Insect(String name, **int** size, String color) {

**super**();

**this**.name = name;

**this**.size = size;

**this**.color = color;

}

//（3）生成成员变量的get和set方法；

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** **int** getSize() {

**return** size;

}

**public** **void** setSize(**int** size) {

**this**.size = size;

}

**public** String getColor() {

**return** color;

}

**public** **void** setColor(String color) {

**this**.color = color;

}

//（4）定义run方法 void run(Insect a)打印输入谁在跑，接收参数为Insect类型；

**void** run(Insect a){

System.***out***.println(a.name +" is running");

}

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

// **TODO** Auto-generated method stub

//（5）在main函数测试该类new一个实例对象i，调用run方法，然后打印输出i的name，size，color

Insect i = **new** Insect("螳螂",3,"绿色");

i.run(i);

System.***out***.println(i.name);

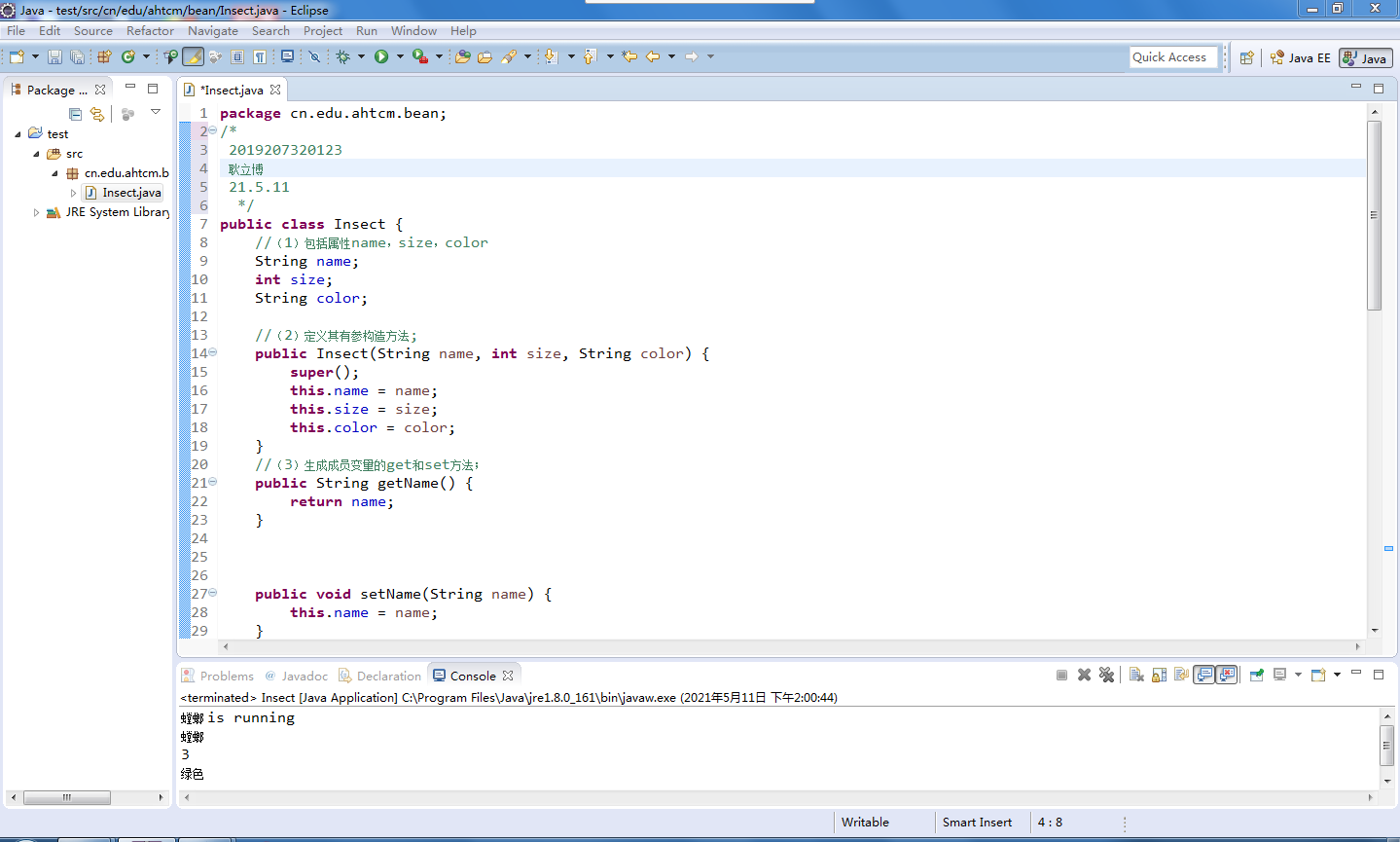
System.***out***.println(i.size);

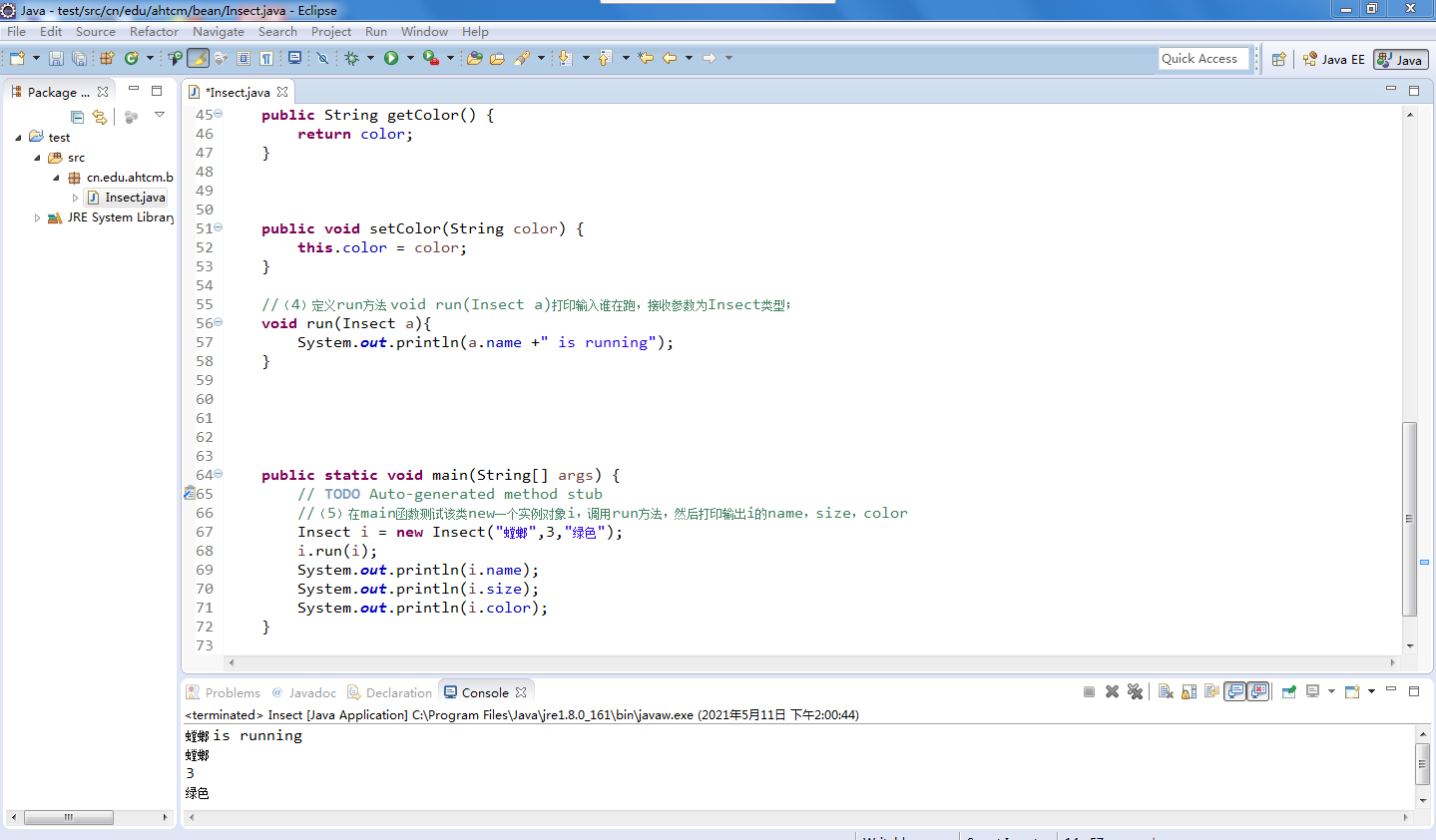
System.***out***.println(i.color);

}

}

**【运行结果】**





**2.实验-2**

**【源码】**

**package** cn.edu.ahtcm.bean;

/\*

2019207320123

耿立博

21.5.11

\*/

**public** **interface** IAttack {

//（1）包含静态的不可改变的字符串常量值ADDRESS字符串，内容为"China";

**public** **final** **static** String ***COUNTRY*** = "China";

//（2）包含两个方法move（）和attack（）

**void** move();

**void** attack();

//（3）包含默认方法default void grow()实现内容为打印输出"Insect grows fastly"

**default** **void** grow(){

System.***out***.println("Insect grows fastly");

}

}

**【运行结果】**

**3.实验-3**

**【源码】**

**package** cn.edu.ahtcm.bean;

/\*

2019207320123

耿立博

21.5.11

\*/

//（1）继承Insect，实现接口IAttack

**public** **class** Ant **extends** Insect **implements** IAttack {

**public** Ant(String name, **int** size, String color) {

**super**(name, size, color);

// **TODO** Auto-generated constructor stub

}

//（2）实现move（）方法：打印输出"Ant move slowly"；

@Override

**public** **void** move() {

// **TODO** Auto-generated method stub

System.***out***.println("Ant move slowly");

}

//（3）实现attack（）方法：打印输出"Ant bite people"；

@Override

**public** **void** attack() {

// **TODO** Auto-generated method stub

System.***out***.println("Ant bite people");

}

//（4）重写父类的run（）方法，打印输出"Ant walk step by step";

@Override

**void** run(Insect a) {

// **TODO** Auto-generated method stub

System.***out***.println("Ant walk step by step");

}

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

// **TODO** Auto-generated method stub

//（5）在main函数new一个Ant实例对象，赋值给对象a，Ant a = new Ant(…)调用move和attack、run、grow方法

Ant a = **new** Ant("蚂蚁a",1,"黑色");

a.move();

a.attack();

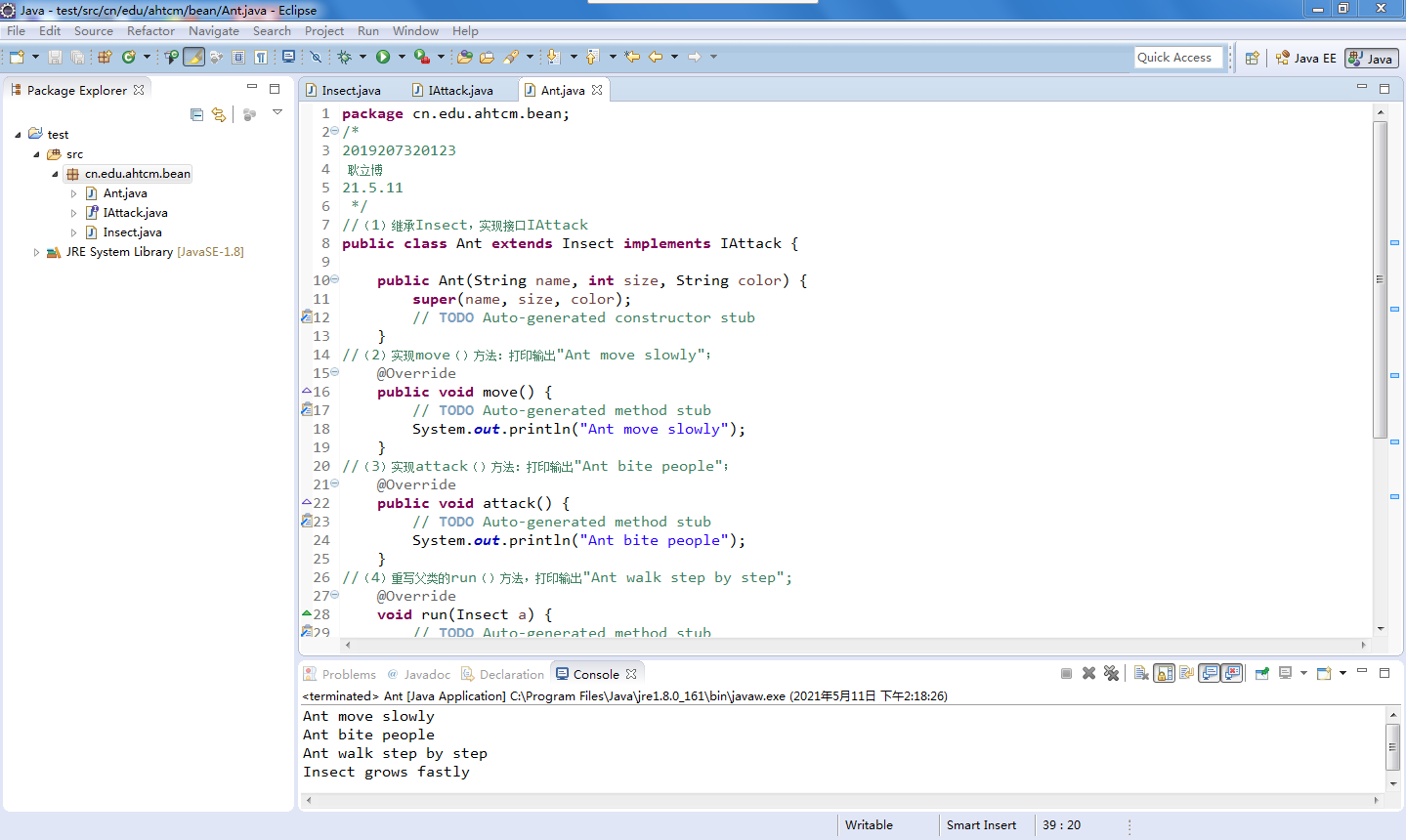
a.run(a);

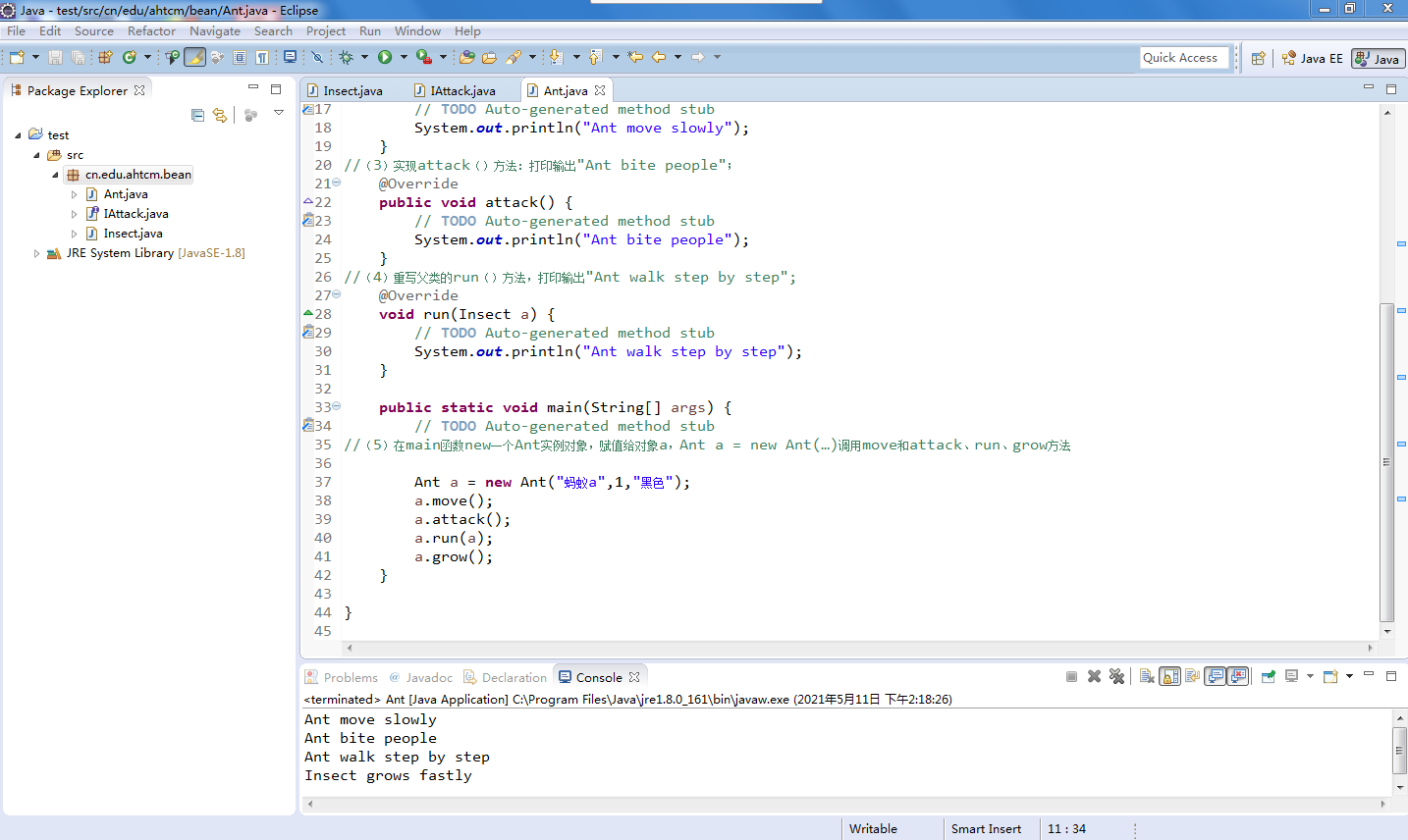
a.grow();

}

}

**【运行结果】**





**4.实验-4**

**【源码】**

**package** cn.edu.ahtcm.bean;

/\*

2019207320123

耿立博

21.5.11

\*/

**public** **abstract** **class** AbsAnimal {

//（1）包括属性name，weight，color;

String name;

**int** size;

String color;

//（2）定义其有参构造方法;

**public** AbsAnimal(String name, **int** size, String color) {

**super**();

**this**.name = name;

**this**.size = size;

**this**.color = color;

}

//（3）定义受保护的run方法，功能是打印输出"Animal run fastly";

**protected** **void** run(){

System.***out***.println("Animal run fastly");

}

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

// **TODO** Auto-generated method stub

//（4）在main方法里实例化抽象对象a,AbsAnimal a = new AbsAnimal(….) ，想一想为什么会报错？如果报错注释掉该语句。

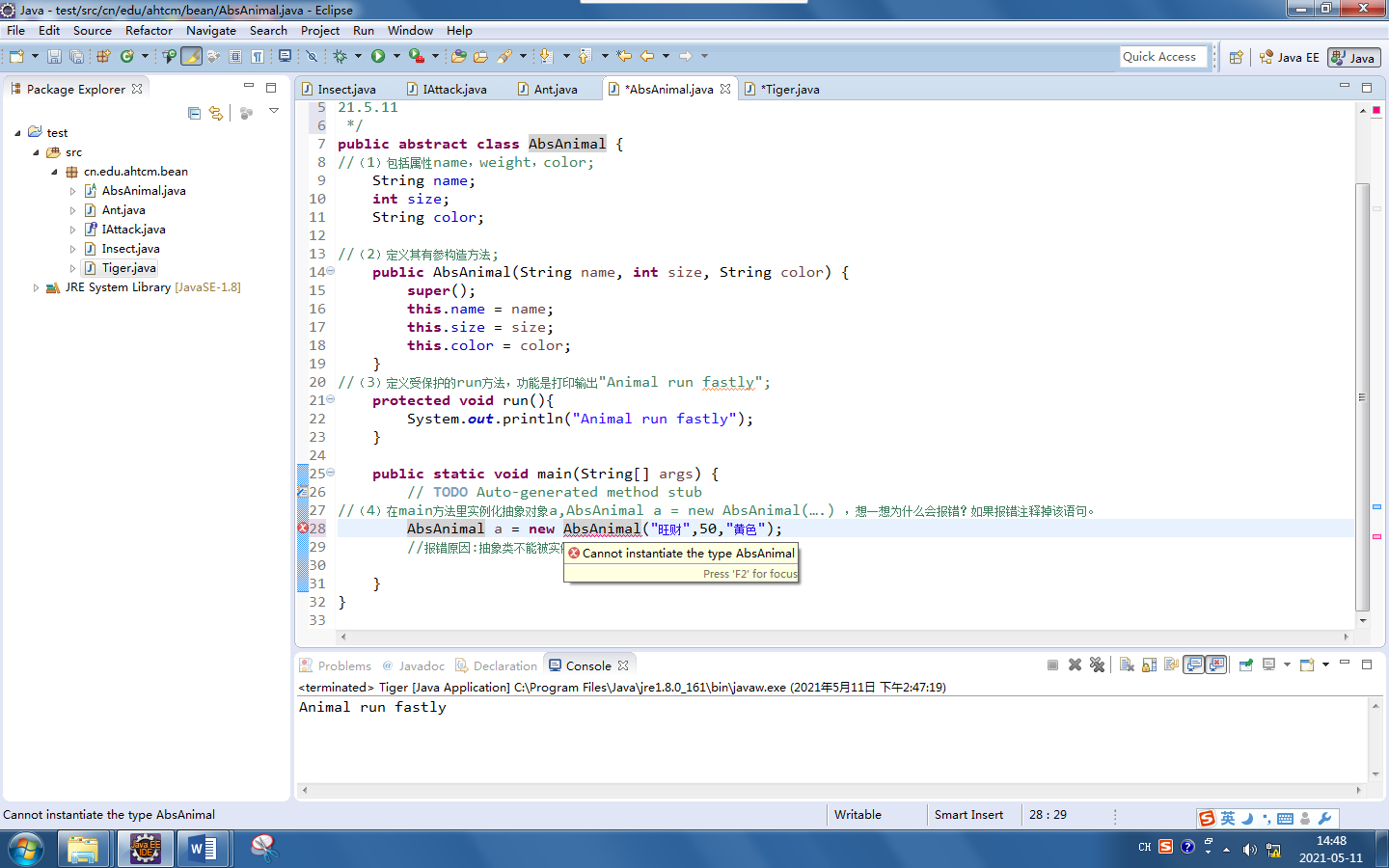
// AbsAnimal a = new AbsAnimal("旺财",50,"黄色");

//报错原因:抽象类不能被实例化只能被继承

}

}

**【运行结果】**



**5.实验-5**

**【源码】**

**package** cn.edu.ahtcm.bean;

/\*

2019207320123

耿立博

21.5.11

\*/

**public** **class** Tiger **extends** AbsAnimal {

**public** Tiger(String name, **int** size, String color) {

**super**(name, size, color);

// **TODO** Auto-generated constructor stub

}

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

// **TODO** Auto-generated method stub

//（1）在main方法中实例化一个Tiger对象

Tiger t = **new** Tiger("泰哥",100,"黄色");

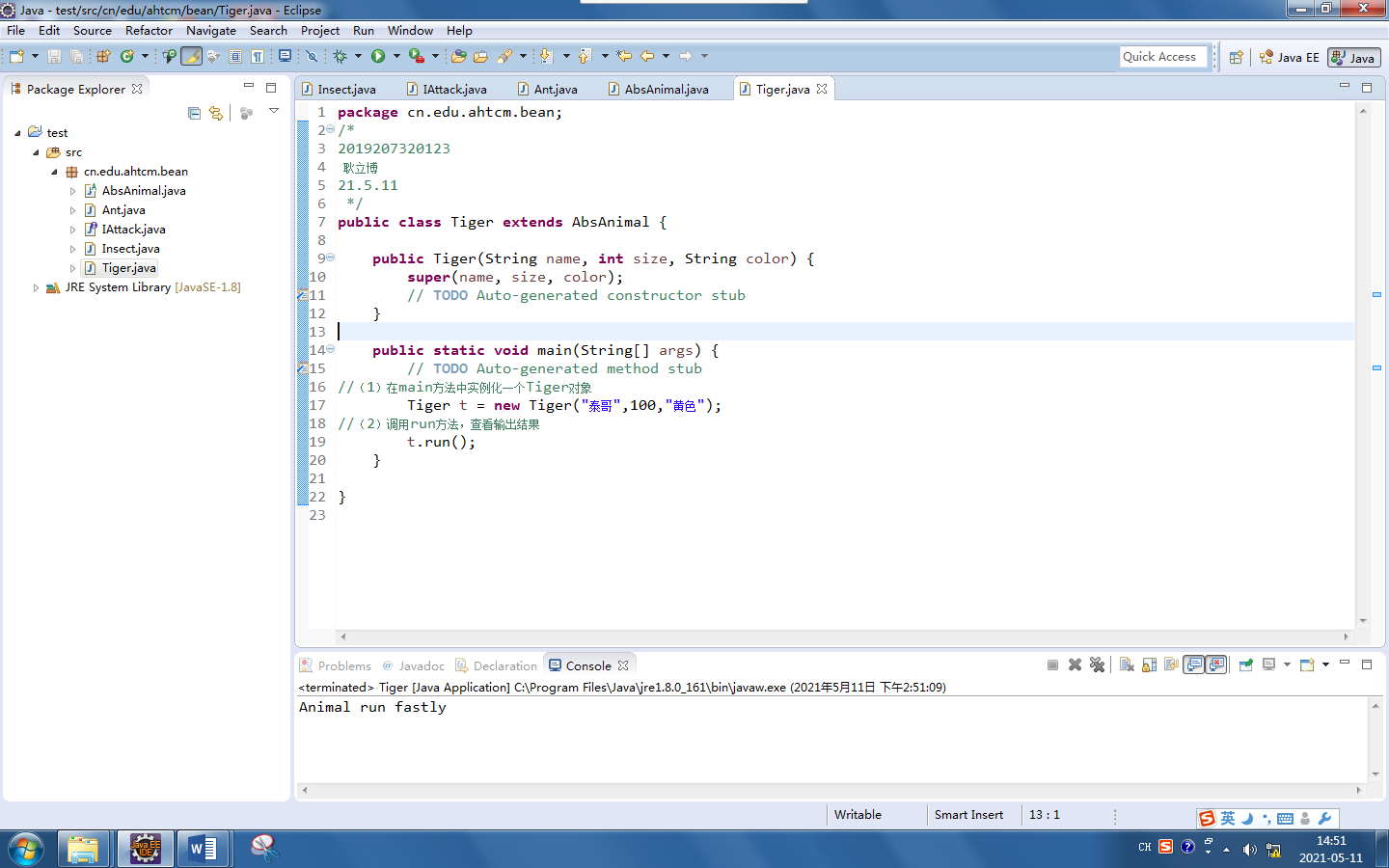
//（2）调用run方法，查看输出结果

t.run();

}

}

**【运行结果】**



**6.实验-6**

**【源码】**

**package** cn.edu.ahtcm.bean;

/\*

2019207320123

耿立博

21.5.11

\*/

**import** java.util.Scanner;

**public** **class** TestDivisionException {

//（1）定义静态方法divisionNum，接收输入的两个整数数，返回整除结果，使用throws抛出ArithmeticException异常；

**public** **static** **double** divisionNum(**int** a,**int** b) **throws** ArithmeticException {

**return** a/b;

}

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

// **TODO** Auto-generated method stub

//（2）main方法里接收键盘输入两个整数a,b;调用divisionNum方法，结果存入变量c，

//（3）如果除数为0，对异常进行捕获；

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

**double** c = 0;

**int** a = in.nextInt();

**int** b = in.nextInt();

**try**{

c = *divisionNum*(a,b);

}**catch**(Exception e){

//（4）打印输出错误信息和错误堆栈信息；

e.printStackTrace();

// System.out.println(e);

}**finally**{

//（5）最后不管是否有错误都打印输出c的结果，并打印“计算完毕”。

System.***out***.println(c);

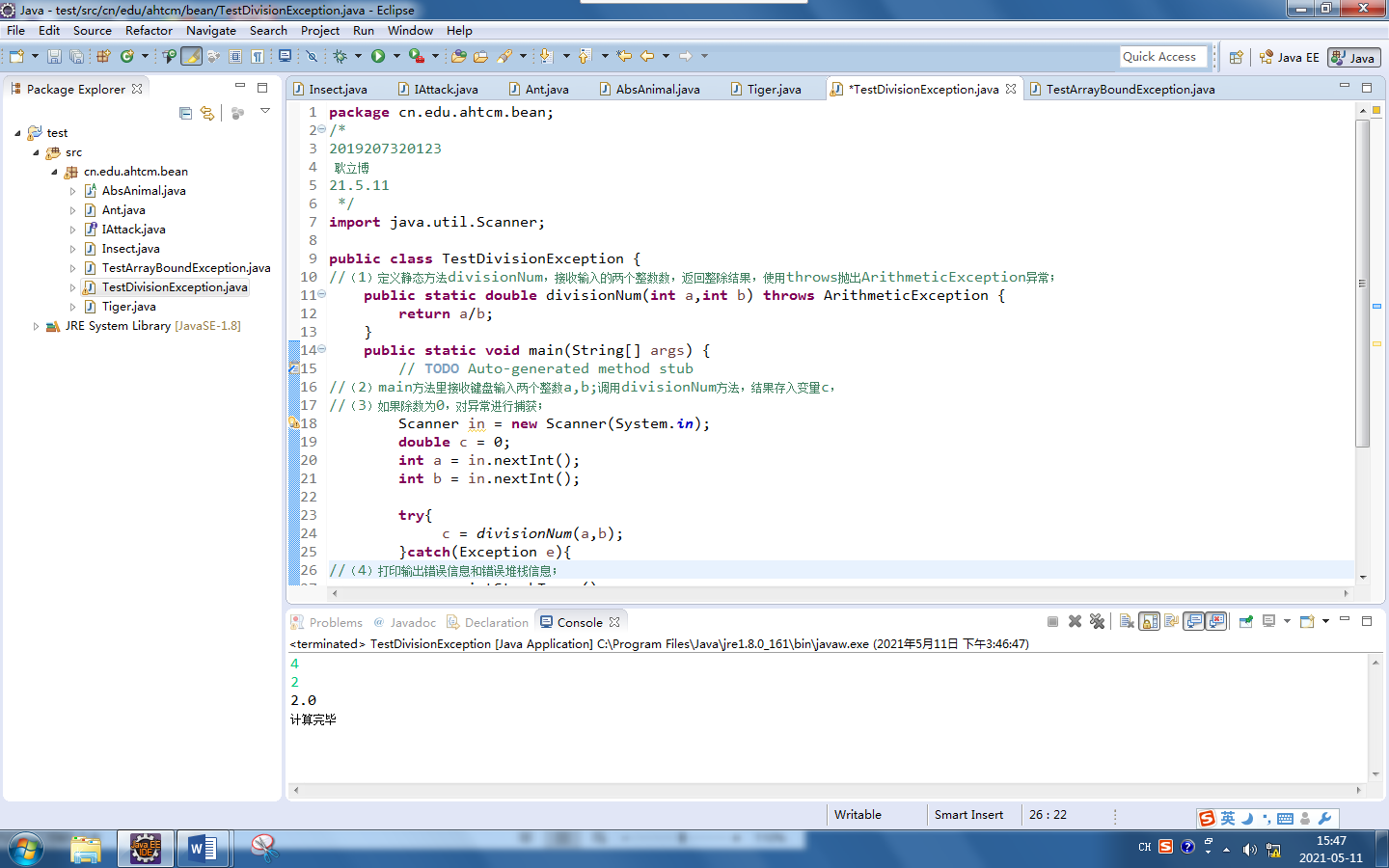
System.***out***.println("计算完毕");

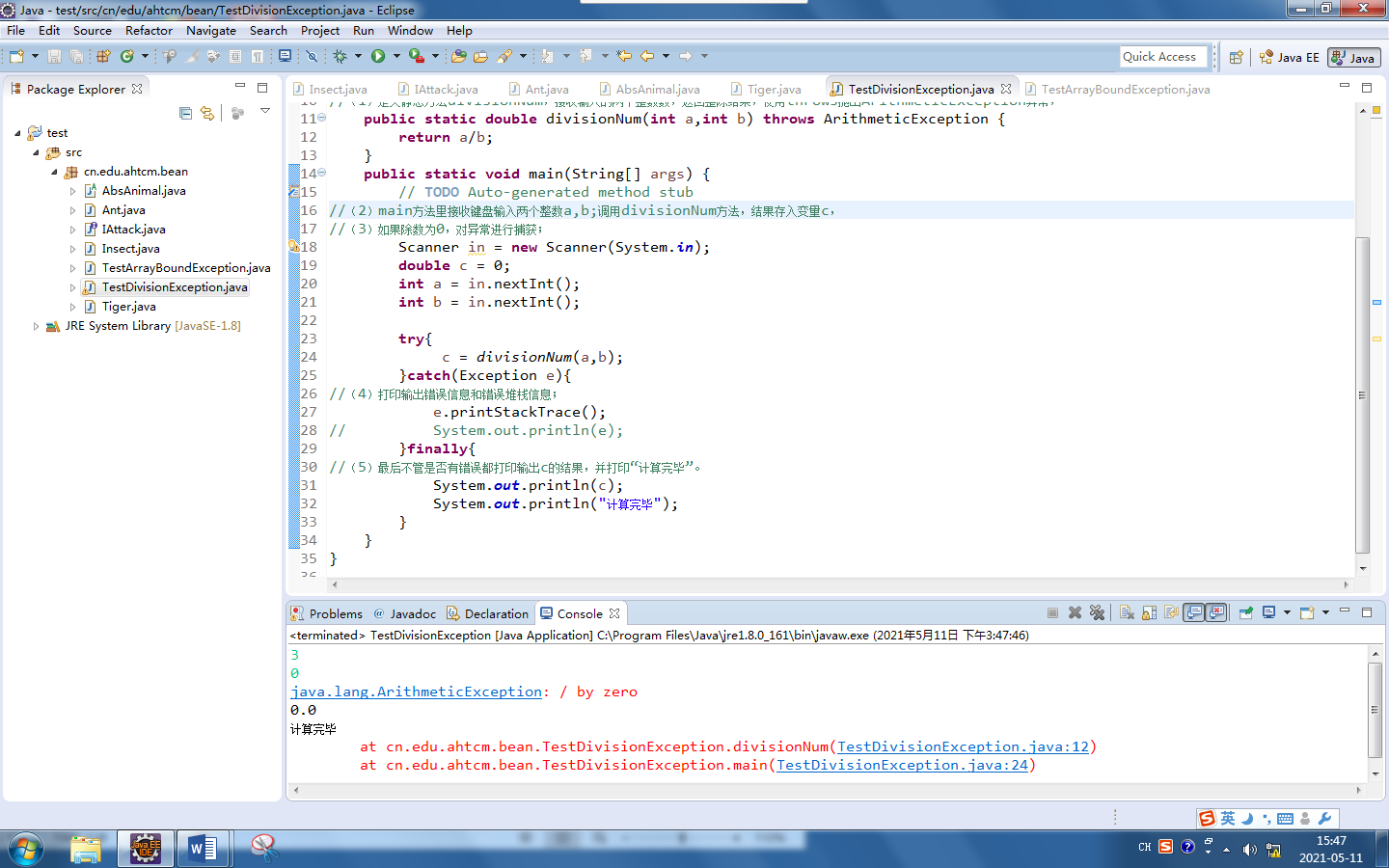
}

}

}

**【运行结果】**





**7.实验-7**

**【源码】**

**package** cn.edu.ahtcm.bean;

/\*

2019207320123

耿立博

21.5.11

\*/

**public** **class** TestArrayBoundException {

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

// **TODO** Auto-generated method stub

//（1）定义字符串数组s，里面存放{“Apple”,”pear”,”banana”}；

String s[] = {"apple","pear","banana"};

//（2）循环打印输出每个元素，对于下标越界进行异常捕获；

**try**{

**for**(**int** i=0;i<4;i++){

System.***out***.println(s[i]);

}

}**catch**(Exception e){

e.printStackTrace();

}**finally**{

//（3）最后不管是否有错误都打印输出“打印完毕”。

System.***out***.println("打印完毕");

}

}

}

**【运行结果】**

