

**成都信息工程大学区块链产业学院**

**《面向对象程序设计（Java）A》**

**上机实验报告**

|  |  |
| --- | --- |
| **实验课程** | **面向对象程序设计（Java）A** |
| **实验项目** | **实验二 Java程序设计基础编程实验** |
| **学生姓名** | **熊灵欣** |
| **学生学号** | **2021131126** |
| **班 级** | **区块链214班** |
| **指导教师** | **李 凡** |
| **实验地点** | **6610实验室** |
| **完成日期** |  |

## 实验名称

实验二 **J**ava程序设计基础编程实验

## 实验目的

通过本实验，学生学习掌握使用Java程序设计基础知识，综合运用基本数据类型、数组等数据类型，以及顺序、循环、分支语句等知识点，设计开发验证性程序。

1. 进一步熟悉使用Eclipse IDE开发Java程序
2. 掌握Java基本数据类型、数组的使用方法
3. 掌握Java分支、循环语句的使用方法
4. 掌握使用For循环对数组进行遍历的程序设计方法
5. 验证使用冒泡排序法对数组进行排序的程序设计方法

## 实验内容

1、设计类PascalTriangle用于表示帕斯卡三角形，要求该类内部使用数组来存放帕斯卡三角形数据，并在构造函数的参数中指定创建的帕斯卡三角形对象的大小（层数），同时编写类PascalTriangleApp，该类通过main方法输出一个8层的帕斯卡三角形，如下所示：

X:\>java PascalTriangleApp

X:\> 1

X:\> 1 1

X:\> 1 2 1

X:\> 1 3 3 1

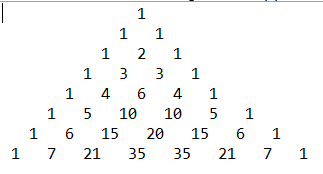
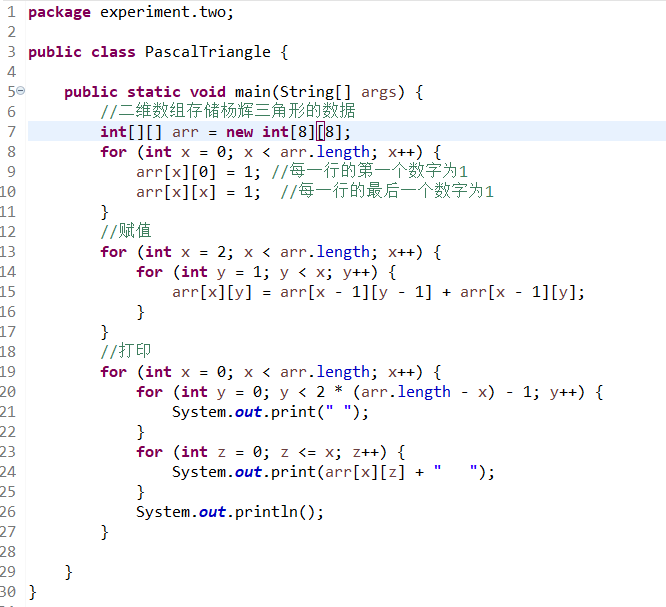
X:\> 1 4 6 4 1

X:\> 1 5 10 10 5 1

X:\> 1 6 15 20 15 6 1

X:\> 1 7 21 35 35 21 7 1

使用Eclipse IDE编写上面的程序，并编译执行，观看运行结果，并附上源程序代码截图和程序执行结果的屏幕截图。



1. 用Java语言编写一个GamePlayer类，可以从命令行重复输入整数，当输入0时，在命令行输出"人物向上移动"；当输入1时，在命令行输出"人物向下移动"；当输入2时，在命令行输出"人物向左移动"；当输入3时，在命令行输出"人物向右移动"；当输入-1时，在命令行输出"退出游戏"，并结束程序运行；当输入其它整数时，在命令行输出"无效输入"。

【参考例程】

**package** experiment.two;

**import** java.util.Scanner;

**public** **class** GamePlayer

{

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

{

//分支语句

//定义一个按键码变量

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

System.***out***.println("启动游戏，请输入人物移动按键码（0：Up 1：Down 2：Left 3：Right）...");

**int** keyCode=-1;

**while**(**true**) {

keyCode=in.nextInt();

**switch**(keyCode)

{

**...... //缺失的程序代码**

}

**if**(keyCode==-1) {

System.***out***.println("退出游戏");

**break**;

}

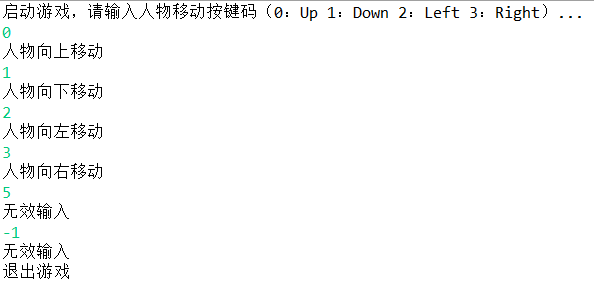
}

}

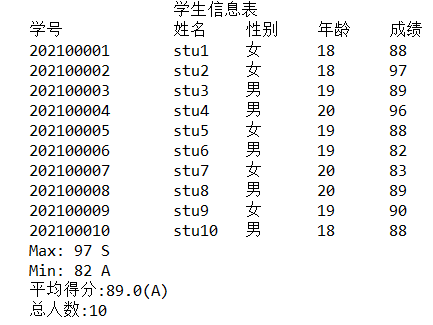
}

使用Eclipse IDE编写上面的程序，补充缺失的程序代码，实现程序的功能要求，并编译执行，观看运行结果，并附上源程序代码截图和程序执行结果的屏幕截图。





3、修改“实验一”中的Student类，程序启动时从命令行输入学生数量，根据输入的学生数量，再分别录入每个学生的学号、姓名、性别、年龄、成绩，再输出每个学生的信息，计算学生的平均分，以及获取学生成绩的最大值与最小值。程序执行输出信息要求如下：



【参考例程】

public class Student {

String sNO,sName,sSex;

int sAge,sJava;

public Student(String num,String name,String sex,int age,int point){//构造方法

sNO = num;

sName = name;

sSex = sex;

sAge = age;

sJava = point;

}

public static char judge(double point){//返回分数对应等级

if(point <= 100 && point >= 90){

return 'S';

}

else if(point < 90 && point >= 80){

return 'A';

}

else if(point < 80 && point >= 70){

return 'B';

}

else if(point < 70 && point >= 60){

return 'C';

}

else{

return 'D';

}

}

public static double getAverageScore(Student arr[]){//计算平均分

double sum = 0;

int i;

for(i = 0;i<arr.length;i++){

sum += arr[i].sJava;

}

return sum / arr.length;

}

public static void getMaxAndMin(Student arr[]){//计算最大值与最小值

int max = -1;

int min = 100;

for(int i = 0;i<arr.length;i++){

if(arr[i].sJava > max)

max = arr[i].sJava;

}

for(int i = 0;i<arr.length;i++){

if(arr[i].sJava < min){

min = arr[i].sJava;

}

}

System.out.println("\tMax: "+max+" "+judge(max));

System.out.println("\tMin: "+min+" "+judge(min));

}

*public static Student[] inputStudentInfo(){* //从命令行录入学生信息与成绩

*...... //缺失的程序代码*

*}*

public String getNO(){

return sNO;

}

public String getName(){

return sName;

}

public String getSex(){

return sSex;

}

public int getAge(){

return sAge;

}

public int getPoint(){

return sJava;

}

public static void main(String[] args) {

//初始化

Student[] student = inputStudentInfo();

//显示信息

System.out.println("\t\t\t学生信息表");

System.out.println("\t学号\t\t姓名\t性别\t年龄\t成绩");

for(int i = 0;i<student.length;i++){

System.out.print("\t"+student[i].getNO()+"\t"+student[i].getName());

System.out.println("\t"+student[i].getSex()+"\t"+student[i].getAge()+"\t"+student[i].getPoint());

}

Student.getMaxAndMin(student);

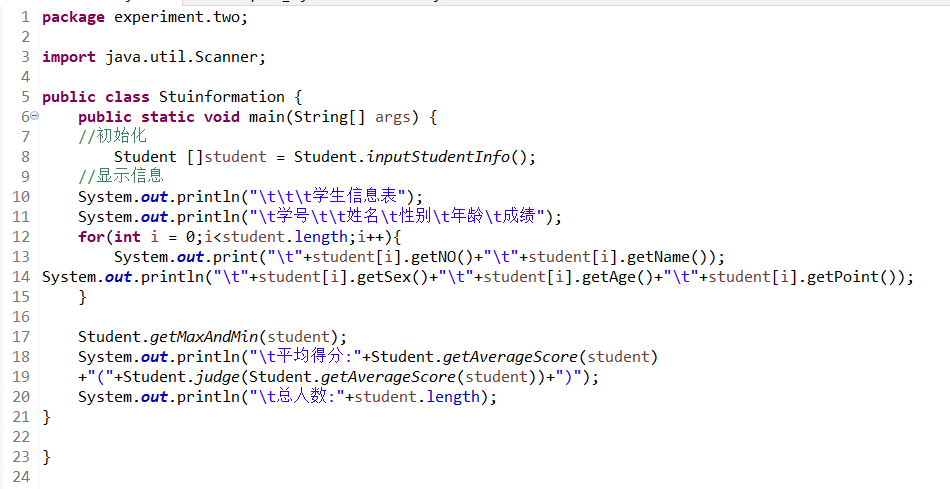
System.out.println("\t平均得分:"+Student.getAverageScore(student) +"("+Student.judge(Student.getAverageScore(student))+")");

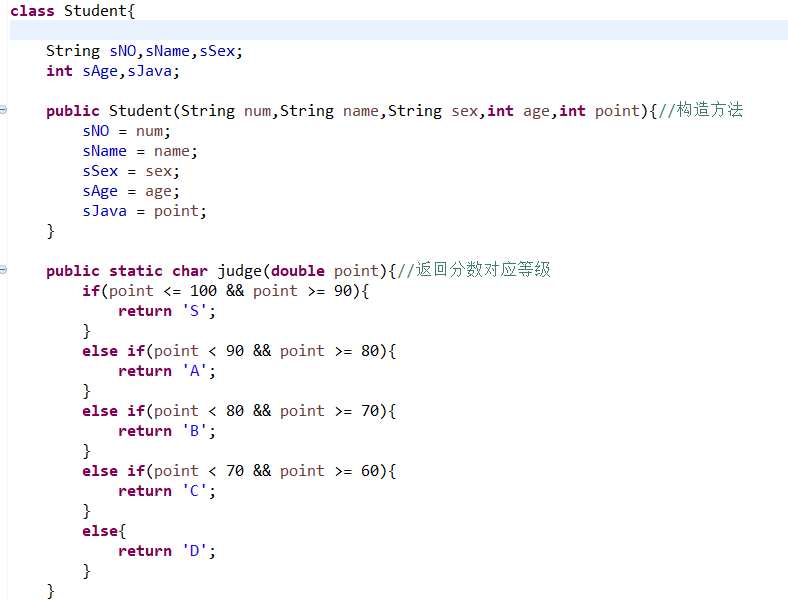
System.out.println("\t总人数:"+student.length);S

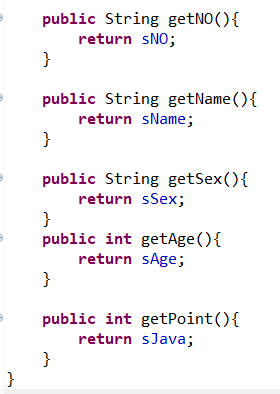
}

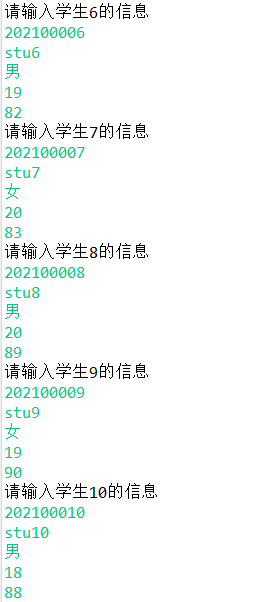
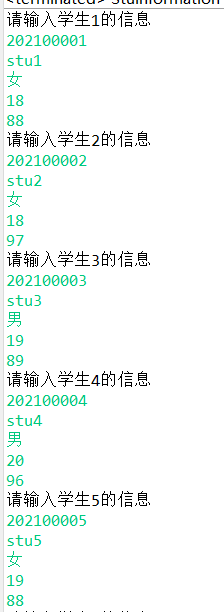
}

根据上面对程序的功能要求，在“参考例程”中补充缺失的程序代码，使用Eclipse IDE编写程序，并编译执行，观看运行结果，并附上源程序代码截图和程序执行结果的屏幕截图。



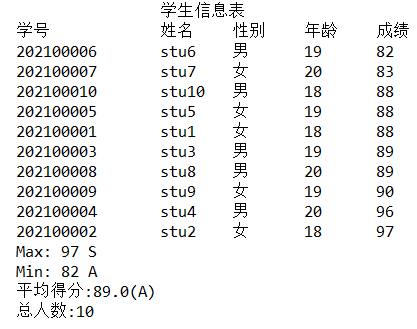








4、继续修改3中的Student类，程序启动时从命令行输入学生数量，根据输入的学生数量，再分别录入每个学生的学号、姓名、性别、年龄、成绩，再使用冒泡排序算法对学生成绩排序后再输出每个学生的信息，计算学生的平均分，以及获取学生成绩的最大值与最小值。程序执行输出信息要求如下：



【参考例程】

public class Student {

String sNO,sName,sSex;

int sAge,sJava;

public Student(String num,String name,String sex,int age,int point){//构造方法

sNO = num;

sName = name;

sSex = sex;

sAge = age;

sJava = point;

}

public static char judge(double point){//返回分数对应等级

if(point <= 100 && point >= 90){

return 'S';

}

else if(point < 90 && point >= 80){

return 'A';

}

else if(point < 80 && point >= 70){

return 'B';

}

else if(point < 70 && point >= 60){

return 'C';

}

else{

return 'D';

}

}

public static double getAverageScore(Student arr[]){//计算平均分

double sum = 0;

int i;

for(i = 0;i<arr.length;i++){

sum += arr[i].sJava;

}

return sum / arr.length;

}

public static void getMaxAndMin(Student arr[]){//计算最大值与最小值

int max = -1;

int min = 100;

for(int i = 0;i<arr.length;i++){

if(arr[i].sJava > max)

max = arr[i].sJava;

}

for(int i = 0;i<arr.length;i++){

if(arr[i].sJava < min){

min = arr[i].sJava;

}

}

System.out.println("\tMax: "+max+" "+judge(max));

System.out.println("\tMin: "+min+" "+judge(min));

}

*public static Student[] inputStudentInfo(){*

*...... //缺失的程序代码*

*}*

*public static void sort(Student[] stus){* //使用冒泡法进行排序

*...... //缺失的程序代码*

*}*

public String getNO(){

return sNO;

}

public String getName(){

return sName;

}

public String getSex(){

return sSex;

}

public int getAge(){

return sAge;

}

public int getPoint(){

return sJava;

}

public static void main(String[] args) {

//初始化

Student[] student = inputStudentInfo(); //录入学生信息与成绩

//显示信息

System.out.println("\t\t\t学生信息表");

System.out.println("\t学号\t\t姓名\t性别\t年龄\t成绩");

sort(student); //根据成绩，对学生信息进行排序

for(int i = 0;i<student.length;i++){

System.out.print("\t"+student[i].getNO()+"\t"+student[i].getName());

System.out.println("\t"+student[i].getSex()+"\t"+student[i].getAge()+"\t"+student[i].getPoint());

}

Student.getMaxAndMin(student);

System.out.println("\t平均得分:"+Student.getAverageScore(student) +"("+Student.judge(Student.getAverageScore(student))+")");

System.out.println("\t总人数:"+student.length);

}

}

根据上面对程序的功能要求，在“参考例程”中补充缺失的程序代码，使用Eclipse IDE编写程序，并编译执行，观看运行结果，并附上源程序代码截图和程序执行结果的屏幕截图。