**public** **static** **double** SumMajorDiagonal (**double** [][]m)

{

**double** sum=0;

**for** (**int** i=0;i<m.length;i++)

{

sum=sum+m[i][i];

}

**return** sum;

}

8.3

**package** test\_6;

**public** **class** SortGrade {

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

// Students' answers to the questions

**char**[][] answers = {

{'A', 'B', 'A', 'C', 'C', 'D', 'E', 'E', 'A', 'D'},

{'D', 'B', 'A', 'B', 'C', 'A', 'E', 'E', 'A', 'D'},

{'E', 'D', 'D', 'A', 'C', 'B', 'E', 'E', 'A', 'D'},

{'C', 'B', 'A', 'E', 'D', 'C', 'E', 'E', 'A', 'D'},

{'A', 'B', 'D', 'C', 'C', 'D', 'E', 'E', 'A', 'D'},

{'B', 'B', 'E', 'C', 'C', 'D', 'E', 'E', 'A', 'D'},

{'B', 'B', 'A', 'C', 'C', 'D', 'E', 'E', 'A', 'D'},

{'E', 'B', 'E', 'C', 'C', 'D', 'E', 'E', 'A', 'D'}};

// Key to the questions

**char**[] keys = {'D', 'B', 'D', 'C', 'C', 'D', 'A', 'E', 'A', 'D'};

**int**[]count=**new** **int**[10];

**int**[]num= {0,1,2,3,4,5,6,7};

// Grade all answers

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

// Grade one student

**int** correctCount = 0;

**for** (**int** j = 0; j < answers[i].length; j++) {

**if** (answers[i][j] == keys[j])

correctCount++;

}

}

count[i]= correctCount;

}

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

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

**if**(count[j]<count[i]) {

**int** tmp1=count[j];

count[j]=count[i];

count[i]=tmp1;

**int** tmp2=num[j];

num[j]=num[i];

num[i]=tmp2;

}

}

}

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

System.***out***.println("Student " + i+ "'s correct count is " +

correctCount);

}

}

}

8.7

**package** test\_6;

**import** java.util.Scanner;

**public** **class** Distance {

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

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

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

System.***out***.print("Enter the number of points: ");

**int** numberOfPoints = input.nextInt();

**int** h=0;

// Create an array to store points

**double**[][] points = **new** **double**[1000][1000];

System.***out***.print("Enter " + numberOfPoints + " points: ");

**while**(numberOfPoints!=h) {

points[h][0]=input.nextDouble();

points[h][1]=input.nextDouble();

points[h][2]=input.nextDouble();

h++;

}

**int** arr=numberOfPoints \*(numberOfPoints -1)/2;

**double**[]result=**new** **double**[arr];

**int**[]x=**new** **int** [arr];

**int**[]y=**new** **int** [arr];

**int** index=0;

**int**[]minIndex=**new** **int**[100];

**for**(**int** i=0;i<number-1;i++)

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

result[index]= compute(points[i],points[j]);

x[index]=i;

y[index]=j;

index++;

}

minIndex=getmin(result);

System.***out***.println("共有"+minIndex[0]+"组点距离最小且相同");

**for**(**int** k=0;k<minIndex[0];k++){

System.***out***.println("第"+(k+1)+"组的两点为："+points[x[minIndex[k+1]]][0]+","+points[x[minIndex[k+1]]][1]+","+points[x[minIndex[k+1]]][2]);

System.***out***.println(" "+points[y[minIndex[k+1]]][0]+","+points[y[minIndex[k+1]]][1]+","+points[y[minIndex[k+1]]][2]);

}

}

}