

Assignment

Q1: Write a Java program to input elements of one dimensional array with size n and define functions to do the following:

اكتب برنامج جافا يقوم بإدخال عناصر مصفوفة ذات بعد واحد (متجه) حجمها n قم بتعريف الدوال الآتية:

1- Print array elements.

طباعة عناصر المصفوفة

2- Count array elements greater than a certain value.

عد عناصر المصفوفة التي تتجاوز عدد معين (val)

3- Calculate the sum of array elements.

حساب مجموع عناصر المصفوفة

4- Calculate the mean.

حساب الوسط الحسابي لعناصر المصفوفة

5- Calculate the variance.

حساب التباين لعناصر المصفوفة

6- Calculate the standard deviation.

حساب الانحراف المعياري لعناصر المصفوفة

7- Calculate the coefficient of variation.

حساب معامل الاختلاف لعناصر المصفوفة

8- Find the max element.

إيجاد القيمة العظمى لعناصر المصفوفة

9- Find the minimum element.

إيجاد القيمة الصغرى لعناصر المصفوفة

10- Calculate the range.

حساب المدى لعناصر المصفوفة

11- Sort array elements.

ترتيب عناصر المصفوفة تصاعديا

12- Calculate the median.

حساب الوسيط لعناصر المصفوفة

```

import java.io.*;
public class Arraystats
{
    // A method to print out array elements
    public static void printA(double[] X)
    {
        for (int i=0; i<X.length; i++)
            System.out.println(X[i]);
    }
    // A method to count array elements > value
    public static int countA(double[] X, double val)
    {
        int count = 0;
        for (int i=0; i<X.length; i++)
        {
            if (X[i] > val)
                count ++;
        }
        return count;
    }
    // A method to sum array elements
    public static double sumA(double[] X)
    {
        double sum = 0;
        for (int i=0; i<X.length; i++)
            sum += X[i];
        return sum;
    }
    // A method to calculate mean
    public static double meanA(double[] X)
    {
        return sumA(X)/X.length;
    }
}

```

استخدام دالة for للتعامل مع المصفوفة ،
أمر طباعة لكل عنصر

استخدام دالة for للتعامل مع المصفوفة ،
تطبيق الشرط على كل عنصر ، عد العنصر إذا
حقق الشرط

قيمة مبدئية للمجموع تساوي صفر
استخدام دالة for للتعامل مع المصفوفة ،
إضافة كل عنصر من المصفوفة إلى المجموع

حساب المتوسط = المجموع / عدد العناصر
هنا استدعاء دالة المجموع واستخدامها في
كتابة دالة المتوسط بدل من كتابة طريقة
جمع العناصر مرة أخرى

// A method to calculate variance

public static double varianceA(**double**[] X)

```
{
    double sum1=0;
    double sum2=0;
    double E1,E2;
    for (int i=0; i<X.length; i++)
    {
        sum1 += X[i];
        sum2 += X[i]*X[i];
    }
    E1 = sum1/X.length;
    E2 = sum2/X.length;
    return E2-Math.pow(E1,2);
}
```

حساب التباين : $Var(X) = E(X^2) - [E(X)]^2$
نستخدم دالة for للتعامل مع المصفوفة
لحساب مجموع العناصر ومجموع مربعات العناصر
وبالتالي حساب متوسط العناصر ومتوسط مربعات
العناصر والتعويض في القانون السابق

// A method to calculate Standard deviation

public static double stddevA(**double**[] X)

```
{
    return Math.sqrt(varianceA(X));
}
```

حساب الانحراف المعياري يساوي الجذر
التربيعي الموجب للتباين

// A method to calculate coefficient of variation

public static double coeffvarA(**double**[] X)

```
{
    return stddevA(X)/meanA(X) *100;
}
```

حساب معامل الاختلاف يساوي
الانحراف المعياري مقسوما على
الوسط الحسابي

// A method to find maximum element

public static double maxA(**double**[] X)

```
{
    double max=X[0];
    for (int i=1; i<X.length; i++)
    {
        if (max<X[i])
            max = X[i];
    }
}
```

نستخدم متغير max يساوي أول عنصر
استخدام دالة for للتعامل مع المصفوفة ،
مقارنة كل عنصر من المصفوفة مع max إذا
كانت قيمة العنصر أكبر نعدل قيمة max

```

    return max;
}
// A method to find minimum element
public static double minA(double[] X)
{
    double min=X[0];
    for (int i=1; i<X.length; i++)
    {
        if (min>X[i])
            min = X[i];
    }
    return min;
}
// A method to calculate range
public static double rangeA(double[] X)
{
    return maxA(X) - minA(X);
}
// A method to sort array elements
public static void sortA(double[] X)
{
    double temp;
    for(int i=0; i<X.length; i++)
        for(int j=i+1; j<X.length; j++)
            if (X[i]>X[j])
            {
                temp = X[i];
                X[i] = X[j];
                X[j] = temp;
            }
}
// A method to calculate the median
public static double medianA(double[] X)
{

```

نستخدم متغير min يساوي أول عنصر
استخدام دالة for للتعامل مع المصفوفة ،
مقارنة كل عنصر من المصفوفة مع min إذا
كانت قيمة العنصر أصغر نعدل قيمة min

حساب المدى يساوي
الفرق بين أكبر قيمة وأصغر قيمة

ترتيب عناصر المصفوفة نستخدم دوارتين الأولى
تتناول كل عنصر من عناصر المصفوفة والثانية تبدأ
من العنصر التالي له وتقارن وتستبدل لتضع القيمة
الأصغر في البداية

```

double m;
int n = X.length;
if (n%2==0)
    m = (X[n/2-1]+X[n/2])/2.0;
else
    m = X[(n-1)/2];
return m;
}

```

حساب الوسيط: إذا كان عدد عناصر المصفوفة زوجي: الوسيط : $m = [X(n/2-1)+X(n/2)]/2$
إذا كان عدد عناصر المصفوفة فردي: الوسيط : $m = X((n-1)/2)$

```

public static void main(String[] args) throws IOException
{
    InputStreamReader inStream = new InputStreamReader( System.in );
    BufferedReader stdin = new BufferedReader( inStream );
    int n = 10;
    double [] arr = new double[n];
    System.out.print("Enter "+n+" array elements: ");
    for(int i=0; i<arr.length;i++)
        arr[i]= Double.parseDouble(stdin.readLine());
    System.out.println("Array data: ");
    printA(arr);
    System.out.println("Array Statistics: ");
    System.out.println("Sum = "+sumA(arr));
    System.out.println("Mean = "+meanA(arr));
    System.out.println("Variance = "+varianceA(arr));
    System.out.println("Std deviation = "+stddevA(arr));
    System.out.println("Coeff. of variation = "+coeffvarA(arr)+"%");
    System.out.println("Max element = "+maxA(arr));
    System.out.println("Min element = "+minA(arr));
    System.out.println("Range = "+rangeA(arr));
    sortA(arr);
    System.out.println("Sorted array data:");
    printA(arr);
    System.out.println("Median = "+medianA(arr));
}
}

```