Name:Kavya Kartha

Class:BE C

Roll No.:325

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

import java.util.\*;

class k\_means

{

static int count1,count2,count3;

static int d[];

static int k[][];

static int tempk[][];

static double m[];

static double diff[];

static int n,p;

static int cal\_diff(int a) // This method will determine the cluster in which an element go at a particular step.

{

int temp1=0;

for(int i=0;i<p;++i)

{

if(a>m[i])

diff[i]=a-m[i];

else

diff[i]=m[i]-a;

}

int val=0;

double temp=diff[0];

for(int i=0;i<p;++i)

{

if(diff[i]<temp)

{

temp=diff[i];

val=i;

}

}//end of for loop

return val;

}

static void cal\_mean() // This method will determine intermediate mean values

{

for(int i=0;i<p;++i)

m[i]=0; // initializing means to 0

int cnt=0;

for(int i=0;i<p;++i)

{

cnt=0;

for(int j=0;j<n-1;++j)

{

if(k[i][j]!=-1)

{

m[i]+=k[i][j];

++cnt;

}}

m[i]=m[i]/cnt;

}

}

static int check1() // This checks if previous k ie. tempk and current k are same.Used as terminating case.

{

for(int i=0;i<p;++i)

for(int j=0;j<n;++j)

if(tempk[i][j]!=k[i][j])

{

return 0;

}

return 1;

}

public static void main(String args[])

{

Scanner scr=new Scanner(System.in);

/\* Accepting number of elements \*/

System.out.println("Enter the number of elements ");

n=scr.nextInt();

d=new int[n];

/\* Accepting elements \*/

System.out.println("Enter "+n+" elements: ");

for(int i=0;i<n;++i)

d[i]=scr.nextInt();

/\* Accepting num of clusters \*/

System.out.println("Enter the number of clusters: ");

p=scr.nextInt();

/\* Initialising arrays \*/

k=new int[p][n];

tempk=new int[p][n];

m=new double[p];

diff=new double[p];

/\* Initializing m \*/

for(int i=0;i<p;++i)

m[i]=d[i];

int temp=0;

int flag=0;

do

{

for(int i=0;i<p;++i)

for(int j=0;j<n;++j)

{

k[i][j]=-1;

}

for(int i=0;i<n;++i) // for loop will cal cal\_diff(int) for every element.

{

temp=cal\_diff(d[i]);

if(temp==0)

k[temp][count1++]=d[i];

else

if(temp==1)

k[temp][count2++]=d[i];

else

if(temp==2)

k[temp][count3++]=d[i];

}

cal\_mean(); // call to method which will calculate mean at this step.

flag=check1(); // check if terminating condition is satisfied.

if(flag!=1)

/\*Take backup of k in tempk so that you can check for equivalence in next step\*/

for(int i=0;i<p;++i)

for(int j=0;j<n;++j)

tempk[i][j]=k[i][j];

System.out.println("\n\nAt this step");

System.out.println("\nValue of clusters");

for(int i=0;i<p;++i)

{

System.out.print("K"+(i+1)+"{ ");

for(int j=0;k[i][j]!=-1 && j<n-1;++j)

System.out.print(k[i][j]+" ");

System.out.println("}");

}//end of for loop

System.out.println("\nValue of m ");

for(int i=0;i<p;++i)

System.out.print("m"+(i+1)+"="+m[i]+" ");

count1=0;count2=0;count3=0;

}

while(flag==0);

System.out.println("\n\n\nThe Final Clusters By Kmeans are as follows: ");

for(int i=0;i<p;++i)

{

System.out.print("K"+(i+1)+"{ ");

for(int j=0;k[i][j]!=-1 && j<n-1;++j)

System.out.print(k[i][j]+" ");

System.out.println("}");

}

}

}

/\*OUTPUT:

student@student-Inspiron-3521:~$ javac k\_means.java

student@student-Inspiron-3521:~$ java k\_means

Enter the number of elements

5

Enter 5 elements:

6 5 7 8 1

Enter the number of clusters:

2

At this step

Value of clusters

K1{ 6 7 8 }

K2{ 5 1 }

Value of m

m1=7.0 m2=3.0

At this step

Value of clusters

K1{ 6 5 7 8 }

K2{ 1 }

Value of m

m1=6.5 m2=1.0

At this step

Value of clusters

K1{ 6 5 7 8 }

K2{ 1 }

Value of m

m1=6.5 m2=1.0

The Final Clusters By Kmeans are as follows:

K1{ 6 5 7 8 }

K2{ 1 }

student@student-Inspiron-3521:~$

\*/