

## DWM Exp1 - 1211061

### Code:

```
import java.util.*;
class KMeans
{
    public static void main(String args[])
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the no. of clusters");
        int k=sc.nextInt();
        System.out.println("Enter the no. of data items");
        int n=sc.nextInt();
        int a[]=new int[n];

        for(int i=0;i<n;i++)
            a[i]=sc.nextInt();
        float m[]=new float[k+5];
        ArrayList k1=new ArrayList();
        ArrayList k2=new ArrayList();
        ArrayList k3=new ArrayList();
        m[0]=10;
        m[1]=20;
        m[2]=3;
        switch(k)
        {
            case 3:
                for(int i=0;i<n;i++)
                {
                    if((Math.abs(a[i]-m[0])>Math.abs(a[i]-
m[1]))&&(Math.abs(a[i]-m[0])>Math.abs(a[i]-m[2])))
                    {
                        if(Math.abs(a[i]-m[2])>Math.abs(a[i]-m[1]))
                            k2.add(a[i]);
                        else
                            k3.add(a[i]);
                    }
                }
                else{
                    if((Math.abs(a[i]-m[1])>Math.abs(a[i]-m[0]))&&(Math.abs(a[i]-
m[1])>Math.abs(a[i]-m[2])))
                    {
                        if(Math.abs(a[i]-m[2])>Math.abs(a[i]-m[0]))
                            k1.add(a[i]);
                        else
```

```

        k3.add(a[i]);
    }
    else
    {
        if((Math.abs(a[i]-m[2])>Math.abs(a[i]-
m[0]))&&(Math.abs(a[i]-m[2])>Math.abs(a[i]-m[1])))
        {
            if(Math.abs(a[i]-m[1])>Math.abs(a[i]-m[0]))
                k1.add(a[i]);
            else
                k2.add(a[i]);
        }
    }
}

for(int i=0;i<k1.size();i++)
    m[3]+=(int)(k1.get(i));
m[3]=m[3]/k1.size();

for(int i=0;i<k2.size();i++)
    m[4]+=(int)(k2.get(i));
m[4]=m[4]/k2.size();

for(int i=0;i<k3.size();i++)
    m[5]+=(int)(k3.get(i));
m[5]=m[5]/k3.size();

System.out.println(m[3]+"\\t"+m[4]+"\\t"+m[5]);
while(m[0]!=m[3]&& m[1]!=m[4]&& m[2]!=m[5])
{
    m[0]=m[3];
    m[1]=m[4];
    m[2]=m[5];
    m[3]=0;
    m[4]=0;
    m[5]=0;
    k1.clear();
    k2.clear();
    k3.clear();
    for(int i=0;i<n;i++)
    {
        if((Math.abs(a[i]-m[0])>Math.abs(a[i]-
m[1]))&&(Math.abs(a[i]-m[0])>Math.abs(a[i]-m[2])))

```

```

        {
            if(Math.abs(a[i]-m[2])>Math.abs(a[i]-m[1]))
                k2.add(a[i]);
            else
                k3.add(a[i]);
        }
    else{
        if((Math.abs(a[i]-m[1])>Math.abs(a[i]-m[0]))&&(Math.abs(a[i]-
m[1])>Math.abs(a[i]-m[2])))
        {
            if(Math.abs(a[i]-m[2])>Math.abs(a[i]-m[0]))
                k1.add(a[i]);
            else
                k3.add(a[i]);
        }
    else
    {
        if((Math.abs(a[i]-m[2])>Math.abs(a[i]-
m[0]))&&(Math.abs(a[i]-m[2])>Math.abs(a[i]-m[1])))
        {
            if(Math.abs(a[i]-m[1])>Math.abs(a[i]-m[0]))
                k1.add(a[i]);
            else
                k2.add(a[i]);
        }
    }
}
}
}

```

```

        System.out.println("Contents of k1: " + k1);
        System.out.println("Contents of k2: " + k2);
        System.out.println("Contents of k3: " + k3);
        for(int i=0;i<k1.size();i++)
            m[3]+=(int)(k1.get(i));

        m[3]=m[3]/k1.size();

        for(int i=0;i<k2.size();i++)
            m[4]+=(int)(k2.get(i));

        m[4]=m[4]/k2.size();
        for(int i=0;i<k3.size();i++)
            m[5]+=(int)(k3.get(i));
        m[5]=m[5]/k3.size();
        System.out.println(m[3]+"\\t"+m[4]+"\\t"+m[5]);
    }
}

```

```

        System.out.println("Contents of k1: " + k1);
System.out.println("Contents of k2: " + k2);
System.out.println("Contents of k3: " + k3);
break;
case 2:
for(int i=0;i<n;i++)
{
    if(Math.abs(a[i]-m[0])>Math.abs(a[i]-m[1]))
        k2.add(a[i]);
    else
        k1.add(a[i]);
}
for(int i=0;i<k1.size();i++)
    m[3]+=(int)(k1.get(i));

    m[3]=m[3]/k1.size();

    for(int i=0;i<k2.size();i++)
    m[4]+=(int)(k2.get(i));
    m[4]=m[4]/k2.size();
System.out.println(m[3]+"\\t"+m[4]);
while(m[0]!=m[3]&& m[1]!=m[4])
{
    m[0]=m[3];
    m[1]=m[4];
    m[3]=0;
    m[4]=0;
    k1.clear();
    k2.clear();

    for(int i=0;i<n;i++)
    {
        if(Math.abs(a[i]-m[0])>Math.abs(a[i]-m[1]))
            k2.add(a[i]);
        else
            k1.add(a[i]);

    }

    System.out.println("Contents of k1: " + k1);
System.out.println("Contents of k2: " + k2);
    for(int i=0;i<k1.size();i++)

```

```

        m[3]+=(int)(k1.get(i));

        m[3]=m[3]/k1.size();

        for(int i=0;i<k2.size();i++)
            m[4]+=(int)(k2.get(i));

        m[4]=m[4]/k2.size();
        System.out.println(m[3]+"\\t"+m[4]);
    }

    System.out.println("Contents of k1: " + k1);
    System.out.println("Contents of k2: " + k2);
    break;
}
}
}

```

## Output:



```

D:\\JCREAT~1\\GE2001.exe
Enter the no. of clusters
2
Enter the no. of data items
9
2 4 10 12 3 20 11 30 25
7.0      25.0
Contents of k1: [2, 4, 10, 12, 3, 11]
Contents of k2: [20, 30, 25]
7.0      25.0
Contents of k1: [2, 4, 10, 12, 3, 11]
Contents of k2: [20, 30, 25]
Press any key to continue..._

```