

Experiment -5B: K-Means 1D Clustering

Source Code:

```
import java.util.*;
import java.lang.Math.*;
class kMeans1D
{
Scanner sc=new Scanner(System.in);
int n,l1,l2;
double m1=0.0,m2=0.0;
int r[]=new int[100];
int r1[]=new int[100];
int r2[]=new int[100];
void accept()
{
System.out.print("Enter number of points:");
n=sc.nextInt();
System.out.print("Enter points:");
for(int i=0;i<n;i++)
r[i]=sc.nextInt();
m1=r[Math.round(n/4)];
m2=r[Math.round(3*n/4)];
l1=0;
l2=0;
for(int i=0;i<n;i++)
if(Math.abs(r[i]-m1)<=Math.abs(r[i]-m2)){
r1[l1]=r[i];
```

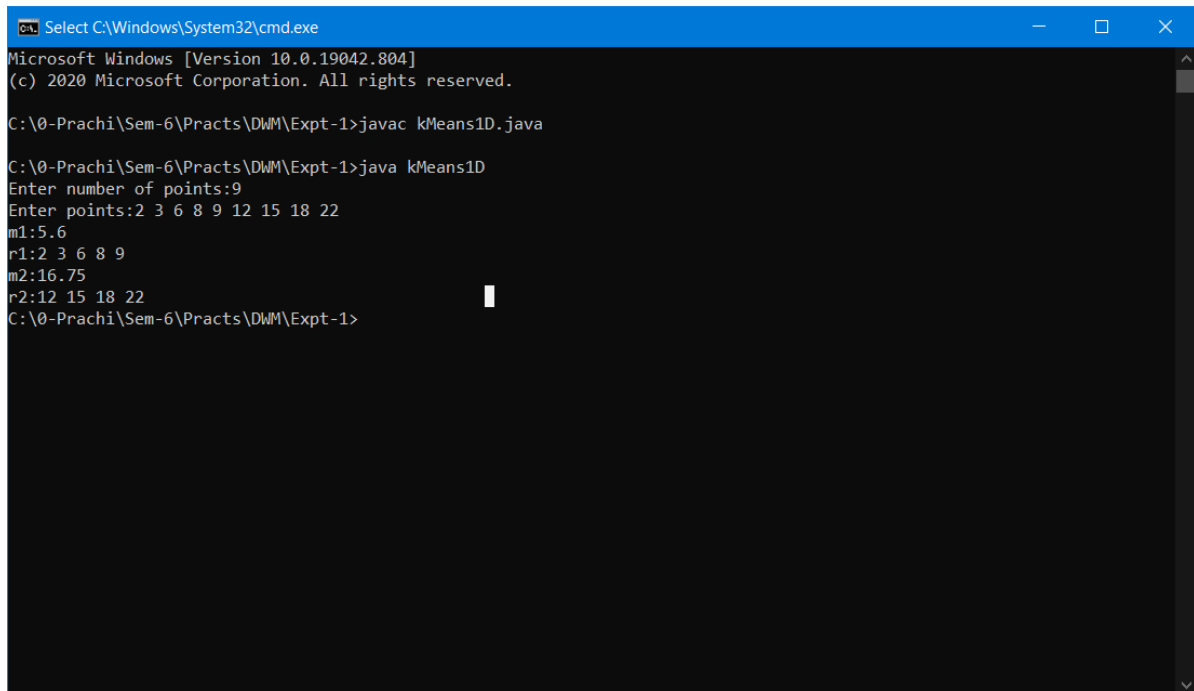
```

l1++;
}
else
{
r2[l2]=r[i];
l2++;
}
}
void calc()
{
m1=0.0;
m2=0.0;
int a=0;
for(int i=0;i<l1;i++)
{
m1+=r1[i];
}
m1/=l1;
for(int i=0;i<l2;i++)
{
m2+=r2[i];
}
m2/=l2;
for(int i=0;i<l2;i++)
{
if(Math.abs(r2[i]-m1)<=Math.abs(r2[i]-m2))
{
r1[l1]=r2[i];a++;
l1++;
l2--;
}
else
break;
}
}

```

```
if(a!=0)
{
for(int i=0;i<12;i++)
{
r2[i]=r2[i+a];
}
calc();
}
}
void disp()
{
System.out.println("m1:"+m1);
System.out.print("r1:");
for(int i=0;i<11;i++)
System.out.print(r1[i]+" ");
System.out.println();
System.out.println("m2:"+m2);
System.out.print("r2:");
for(int i=0;i<12;i++)
System.out.print(r2[i]+" ");
}
public static void main(String args[])
{
kMeans1D k=new kMeans1D();
k.accept();
k.calc();
k.disp();
}
}
```

Output:



```

Select C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19042.804]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\0-Prachi\Sem-6\Practs\DWM\Expt-1>javac kMeans1D.java

C:\0-Prachi\Sem-6\Practs\DWM\Expt-1>java kMeans1D
Enter number of points:9
Enter points:2 3 6 8 9 12 15 18 22
m1:5.6
r1:2 3 6 8 9
m2:16.75
r2:12 15 18 22
C:\0-Prachi\Sem-6\Practs\DWM\Expt-1>
```