**Practical – 1**

**Write a program to perform data smoothing by binning.**

import java.util.Arrays;

import java.util.Scanner;

public class smoothing {

static int a = 0;

public static void main(String args[]) {

smoothing sm = new smoothing();

//scanning size

Scanner sc = new Scanner(System.in);

System.out.println("enter the size of array:");

int size = sc.nextInt();

//taking element of array

int[] data = new int[size];

System.out.println("enter elemeant of array:");

for (int i = 0; i < size; i++) {

data[i] = sc.nextInt();

}

//array sorting

System.out.println("sorted array:");

Arrays.sort(data);

for (int i = 0; i < size; i++) {

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

}

System.out.println();

//bin depth

System.out.println("enter the bin depth:");

int bin\_depth = sc.nextInt();

//bin number

int bin\_number = data.length / bin\_depth;

//creating bin

int[][] bin = new int[bin\_number][bin\_depth];

for (int i = 0; i < bin\_number; i++) {

for (int j = 0; j < bin\_depth; j++) {

bin[i][j] = data[a];

a++;

}

}

//displying bin

for (int i = 0; i < bin\_number; i++) {

System.out.print("bin" + (i + 1) + "--> ");

for (int j = 0; j < bin\_depth; j++) {

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

}

System.out.println();

}

//creating mean array

int[][] mean=new int[bin\_number][bin\_depth];

for (int i = 0; i < bin\_number; i++) {

System.arraycopy(bin[i], 0, mean[i], 0, bin\_depth);

System.out.println();

}

//calling mean fuction for mean smoothing

sm.mean(bin\_number, bin\_depth, mean);

//creating median array

int[][] median=new int[bin\_number][bin\_depth];

for (int i = 0; i < bin\_number; i++) {

System.arraycopy(bin[i], 0, median[i], 0, bin\_depth);

System.out.println();

}

//calling mean fuction for median smoothing

sm.median(bin\_number, bin\_depth, median);

//creating median array

int[][] boundary=new int[bin\_number][bin\_depth];

for (int i = 0; i < bin\_number; i++) {

System.arraycopy(bin[i], 0, boundary[i], 0, bin\_depth);

System.out.println();

}

//creating median array

sm.boundary(bin\_number, bin\_depth, boundary);

}

//mean

void mean(int bin\_number, int bin\_depth, int[][] mean) {

String s = "mean";

double sum = 0;

int sum1=0;

for (int i = 0; i < bin\_number; i++) {

for (int j = 0; j < bin\_depth; j++) {

sum = sum + mean[i][j];

}

sum = sum / bin\_depth;

sum1=(int)sum;

if((sum-sum1)>0.50)

sum1++;

for (int j = 0; j < bin\_depth; j++) {

mean[i][j] = sum1;

}

sum = 0;

sum1= 0;

//4, 8, 9, 15, 21, 21, 24, 25, 26, 28, 29, 34

}

display(bin\_number, bin\_depth, mean, s);

}

//median

void median(int bin\_number, int bin\_depth, int[][] median) {

String s = "median";

int bin\_depth1 = bin\_depth / 2;

for (int i = 0; i < bin\_number; i++) {

if (bin\_depth1 % 2 == 0) {

int replace = (median[i][bin\_depth1] + median[i][bin\_depth1 - 1]) / 2;

for (int j = 0; j < bin\_depth; j++) {

median[i][j] = replace;

}

} else {

for (int j = 0; j < bin\_depth; j++) {

median[i][j] = median[i][bin\_depth1];

}

}

}

display(bin\_number, bin\_depth, median, s);

}

//displaying bins after smoothing

void display(int bin\_number, int bin\_depth, int[][] normal, String s) {

System.out.println("bins after " + s + " smoothing");

for (int i = 0; i < bin\_number; i++) {

System.out.print("bin" + (i + 1) + "--> ");

for (int j = 0; j < bin\_depth; j++) {

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

}

System.out.println();

}

}

//boundary

void boundary(int bin\_number, int bin\_depth, int[][] boundary) {

String s = "boundary";

int min1;

int min2;

for (int i = 0; i < bin\_number; i++) {

for (int j = 0; j < bin\_depth - 2; j++) {

if ((boundary[i][j+1]-boundary[i][j]) < (boundary[i][j + 2]-boundary[i][j + 1])) {

min1 = i;

min2 = j;

} else {

min1 = i;

min2 = j + 1;

}

for (j = 1; j <= bin\_depth - 2; j++) {

boundary[i][j] = boundary[min1][min2];

}

}

}

display(bin\_number, bin\_depth, boundary, s);

}

}

**Output:-**

