package kmeans2;

import static java.lang.Math.abs;

import java.util.ArrayList;

import java.util.Collections;

import java.util.Scanner;

public class kmeans2 {

int k;

int noOfItems;

ArrayList<Integer> dataItems;

ArrayList<Integer> cz;

ArrayList<Integer> oldCz;

ArrayList<Integer> row;

ArrayList<ArrayList<Integer>> groups;

Scanner input;

public kmeans2(int k, int noOfItems) {

this.k = k;

this.noOfItems = noOfItems;

dataItems = new ArrayList<>();

cz = new ArrayList<>();

oldCz = new ArrayList<>();

row = new ArrayList<>();

groups = new ArrayList<>();

input = new Scanner(System.in);

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

groups.add(new ArrayList<>());

}

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

System.out.println("Enter Value for: " + (i + 1) + " item");

dataItems.add(input.nextInt());

if (i < k) {

cz.add(dataItems.get(i));

System.out.println("C" + (i + 1) + " is " + cz.get(i));

}

}

int iter = 1;

do {

for (int aItem : dataItems) {

for (int c : cz) {

row.add(abs(c - aItem));

}

groups.get(row.indexOf(Collections.min(row))).add(aItem);

row.removeAll(row);

}

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

if (iter == 1) {

oldCz.add(cz.get(i));

} else {

oldCz.set(i, cz.get(i));

}

if (!groups.get(i).isEmpty()) {

cz.set(i, average(groups.get(i)));

}

}

if (!cz.equals(oldCz)) {

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

groups.get(i).removeAll(groups.get(i));

}

}

iter++;

} while (!cz.equals(oldCz));

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

System.out.println("New C" + (i + 1) + " " + cz.get(i));

}

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

System.out.println("Group " + (i + 1));

System.out.println(groups.get(i).toString());

}

System.out.println("Number of Itrations: " + iter);

}

public static void main(String[] args) {

Scanner c = new Scanner(System.in);

System.out.println("Enter Value of K");

int k = c.nextInt();

System.out.println("Enter No of Data Items");

int noOfItems = c.nextInt();

new kmeans2(k, noOfItems);

}

public static int average(ArrayList<Integer> list) {

int sum = 0;

for (Integer value : list) {

sum = sum + value;

}

return sum / list.size();

}

}