



Day 0: Mean, Median, and Mode

 by AvmnuSng

Problem

Submissions

Leaderboard

Discussions

Editorial

Tutorial

Objective

In this challenge, we practice calculating the *mean*, *median*, and *mode*. Check out the [Tutorial](#) tab for learning materials and an instructional video!

Task

Given an array, X , of N integers, calculate and print the respective *mean*, *median*, and *mode* on separate lines. If your array contains more than one *modal value*, choose the numerically smallest one.

Note: Other than the modal value (which will always be an integer), your answers should be in decimal form, rounded to a scale of **1** decimal place (i.e., **12.3**, **7.0** format).

Input Format

The first line contains an integer, N , denoting the number of elements in the array.
The second line contains N space-separated integers describing the array's elements.

Constraints

- $10 \leq N \leq 2500$
- $0 < x_i \leq 10^5$, where x_i is the i^{th} element of the array.

Output Format

Print **3** lines of output in the following order:

- Print the *mean* on a new line, to a scale of **1** decimal place (i.e., **12.3**, **7.0**).
- Print the *median* on a new line, to a scale of **1** decimal place (i.e., **12.3**, **7.0**).
- Print the *mode* on a new line; if more than one such value exists, print the numerically smallest one.

Sample Input

```
10
64630 11735 14216 99233 14470 4978 73429 38120 51135 67060
```

Sample Output

```
43900.6
44627.5
4978
```

Explanation

Mean:

We sum all N elements in the array, divide the sum by N , and print our result on a new line.

$$\mu = \frac{x_0 + x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8 + x_9}{10} = \frac{439006}{10} = 43900.6$$

Median:

To calculate the median, we need the elements of the array to be sorted in either non-increasing or non-decreasing order. The sorted array $X = \{4978, 11735, 14216, 14470, 38120, 51135, 64630, 67060, 73429, 99233\}$. We then average the two middle elements:

$$\text{median} = \frac{x_4 + x_5}{2} = \frac{89255}{2} = 44627.5$$

and print our result on a new line.

Mode:

We can find the number of occurrences of all the elements in the array:

```
4978 : 1
11735 : 1
14216 : 1
14470 : 1
38120 : 1
51135 : 1
64630 : 1
67060 : 1
73429 : 1
99233 : 1
```

Every number occurs once, making **1** the maximum number of occurrences for any number in X . Because we have multiple values to choose from, we want to select the smallest one, **4978**, and print it on a new line.

[f](#) [t](#) [in](#)

Solved score: 30.00pts

Submissions: 21057

Max Score: 30

Difficulty: Easy

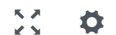
Rate This Challenge:

☆☆☆☆☆

[More](#)

Current Buffer (saved locally, editable)  

Java 8



```
1 import java.io.*;
2 import java.util.*;
3
4 public class Solution {
5
6     public static void main(String[] args) {
7         /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution. */
8         Scanner scan = new Scanner(System.in);
9         int size = scan.nextInt();
10        int arr[] = new int[size];
11        double sum = 0;
12        Map<Integer, Integer> map = new HashMap<>();
13        for(int i=0; i<size; i++) {
14            arr[i] = scan.nextInt();
15            sum += arr[i];
16            if(map.containsKey(arr[i])) {
17                map.put(arr[i], map.get(arr[i])+1);
18            }
19            else
20                map.put(arr[i], 1);
21        }
22        scan.close();
23
24        Arrays.sort(arr);
25        double mean = getMean(arr);
26        double median = getMedian(arr);
27        int mode = getMode(arr);
28        System.out.println(mean + "\n" + median + "\n" + mode);
29    }
}
```

```

30     }
31
32     private static double getMean(int[] arr) {
33         double sum = 0;
34         for (int value : arr)
35             sum += value;
36         return sum / arr.length;
37     }
38
39     private static double getMedian(int[] arr) {
40         int size = arr.length;
41         return (size % 2 == 0) ? ((arr[size / 2 - 1] + arr[size / 2]) / (double) 2) : arr[size / 2];
42     }
43
44     private static int getMode(int[] arr) {
45         int modeCount = 0, mode = 0, currCount = 0;
46
47         for (int candidateMode : arr) {
48             currCount = 0;
49
50             for (int element : arr) {
51                 if (candidateMode == element)
52                     currCount++;
53             }
54
55             if (currCount > modeCount) {
56                 modeCount = currCount;
57                 mode = candidateMode;
58             }
59         }
60         return mode;
61     }
62 }

```

Line: 1 Col: 1

 [Upload Code as File](#) ☐ Test against custom input

Run Code

Submit Code

Join us on IRC at [#hackerrank](#) on freenode for hugs or bugs.

[Contest Calendar](#) | [Blog](#) | [Scoring](#) | [Environment](#) | [FAQ](#) | [About Us](#) | [Support](#) | [Careers](#) | [Terms Of Service](#) | [Privacy Policy](#) | [Request a Feature](#)