Problem C

A mid-summer night's dream

Input: standard input

Output: standard output

This is year 2200AD. Science has progressed a lot in two hundred years. Two hundred years is mentioned here because this problem is being sent back to 2000AD with the help of time machine. Now it is possible to establish direct connection between man and computer CPU. People can watch other people's dream on 3D displayer (That is the monitor today) as if they were watching a movie. One problem in this century is that people have become so dependent on computers that their analytical ability is approaching zero. Computers can now read problems and solve them automatically. But they can solve only difficult problems. There are no easy problems now. Our chief scientist is in great trouble as he has forgotten the number of his combination lock. For security reasons computers today cannot solve combination lock related problems. In a mid-summer night the scientist has a dream where he sees a lot of unsigned integer numbers flying around. He records them with the help of his computer, Then he has a clue that if the numbers are (X1, X2, … , Xn) he will have to find an integer number **A** (This **A** is the combination lock code) such that

(|X1-A| + |X2-A| + … … + |Xn-A|) is minimum.

Input

Input will contain several blocks. Each block will start with a number n (0<n<=1000000) indicating how many numbers he saw in the dream. Next there will be n numbers. All the numbers will be less that 65536. The input will be terminated by end of file.

Output

For each set of input there will be one line of output. That line will contain the number A (If there are more than one such value print the smallest) and how many such numbers are there in the input and then how many such different integer values are there (the values need not be present in the input). These numbers will be seperated by single space.

Sample Input:

```
2
10
10
4
1
```

2
2
4

Sample Output:

10 2 1
2 2 1

Shahriar Manzoor
16-12-2000