UVA10057

題目

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 peoples 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.

今年是公元2200年。在過去的200年中，科學取得了很大進步。這裡提到了兩百年，因為這個問題是在時光機的幫助下被發送回公元2000年的。現在可以在人與電腦之間建立直接連接。人們可以在3D顯示器上觀看別人的夢，就像在看電影一樣。本世紀最大的一個問題是，人們對電腦的依賴性變得如此之高，以至於他們的分析能力接近於零。現在，電腦可以讀取問題並自動解決，但是他們只能解決困難的問題(現在已經沒有簡單的問題了)。我們的首席科學家遇到了很大的麻煩，因為他忘記了密碼。出於安全原因，當今的電腦無法解決與密碼相關的問題。

題目

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

$$(|X1 - A| + |X2 - A| + \ldots + |Xn - A|)$$

is minimum.

在仲夏夜裡，科學家做了一個夢，在那裡他看到許多無號整數飛來飛去。
他在電腦的幫助下記錄了它們，然後他知道如果數字為(X1，X2，...，Xn)。
他需要找到一個整數A(此A為密碼)，使得能夠得到以下式子的最小值。

$$(|X1 - A| + |X2 - A| + \ldots + |Xn - A|)$$

輸入與輸出

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.

輸入：輸入包含多組測資。每組測資第一行為數字n (0 < n ≤ 1000000)，表示他在夢中看到了多少個數字。接下來有n個數字，所有數字都小於65536。

Output：For each set of input there will be one line of output. That line will contain the minimum possible value for A. Next it will contain how many numbers are there in the input that satisfy the property of A (The summation of absolute deviation from A is minimum). And finally you have to print how many possible different integer values are there for A (these values need not be present in the input). These numbers will be separated by single space.

輸出：對於每組測資，輸出三個整數。第一個數字是能得到該算式最小值的A。第二個數字是|Xi − A|為最小值的數量。第三行數字是可能有幾種最小值。

範例測資

Input	*Output*
2 10 10 4 1 2 2 4	10 2 1 2 2 1

程式碼說明

Step 1：輸入測資

```
 6      int n;
 7      while(cin>>n){
 8          vector<int> v(n);
 9          for(int &i:v){
10              cin>>i;
11          }
```

已宣告變數	註解
n	有幾個數字
v	n個數字

程式碼說明

Step 2：排列數字並求答案

```
12          sort(v.begin(),v.end());
13          if(n%2==0){
14              cout<<v[n/2-1]<<" ";
15              int num=0;
16              for(int i:v)
17                  if(i==v[n/2-1]||i==v[n/2])
18                      num++;
19              cout<<num<<" "<<v[n/2]-v[n/2-1]+1<<endl;
20
21          }else{
22              cout<<v[n/2]<<" ";
23              int num=0;
24              for(int i:v)
25                  if(i==v[n/2])
26                      num++;
27              cout<<num<<" "<<1<<endl;;
```

已宣告變數	註解
n	有幾個數字
v	n個數字
num	|Xi-A|最小值

完整程式碼

```cpp
#include<iostream>
#include<vector>
#include<algorithm>
using namespace std;
int main(){
    int n;
    while(cin>>n){
        vector<int> v(n);
        for(int &i:v){
            cin>>i;
        }
        sort(v.begin(),v.end());
        if(n%2==0){
            cout<<v[n/2-1]<<" ";
            int num=0;
            for(int i:v)
                if(i==v[n/2-1]||i==v[n/2])
                    num++;
            cout<<num<<" "<<v[n/2]-v[n/2-1]+1<<endl;

        }else{
            cout<<v[n/2]<<" ";
            int num=0;
            for(int i:v)
                if(i==v[n/2])
                    num++;
            cout<<num<<" "<<1<<endl;;
        }
```

資料來源

英文題目：
https://vjudge.net/problem/UVA-10057

中文題目：
https://zerojudge.tw/ShowProblem?problemid=e606

Thank You