

PDP-28 (2016) - Camp common - 2 (bookeaters)

Time limit: 1.0s Memory limit: 64M

BOOKEATERS

In a school, there are N children. The peculiar thing about this school is that all children are *book-eaters*: they really enjoy reading books. Now that the summer holidays are coming, the children are eager to read new books in their free time. Let's assume that each child has a new book that neither they nor any other child has read. In order for the children to read as many new books as possible, they decide to split into groups and spend their holidays together. If a group consists of K children, then there will be K new books in it, and each child will read all of them before autumn. For the children, the larger the groups, the better!

The children present their idea to the school principal, who finds it amazing and decides to help them make it happen. However, he quickly sees a difficulty that the children have not foreseen. It's not easy to find rooms to accommodate very large groups during the summer. The principal quickly realizes that, in order to implement their plan, the children will need to be divided into as many groups as possible. However, the children are upset because this means they will read fewer books. In an attempt to find a solution, the principal asks each child how many books they would be satisfied with reading this summer.

Write a program that, based on the children's responses, calculates the maximum possible number of groups that can be formed so that all children are satisfied. If there are multiple ways to divide the children into the maximum possible number of groups, find the one that minimizes the number of children in the largest group.

Of course, it's understood that each group must include at least one child and that each child will belong to exactly one group.

Input

The first line of input will contain a natural number N : the number of children in the school. The next N lines will each contain exactly one natural number, $B_1, \dots, B_N$: the responses of the children. That is, the i -th child asked by the principal answered that they would be satisfied if they read at least B_i books this summer.

Output

The output must consist of one line containing two natural numbers M and K , separated by a single space. M will be the maximum possible number of groups that can be formed so that all children are satisfied. For this maximum M , K will be the minimum number of children that the largest group can have.

Constraints

$$1 \leq N \leq 1.000.000$$

$$1 \leq B_i \leq N$$

Execution time limit: 1 sec.

Memory limit: 64 MB.

Example of Input - Output:

STDIN (bookeaters.in)

5
2
1
2
2
3

STDOUT (book eaters.out)

2 3

Explanation: The maximum number of groups that can be formed is $M = 2$. (There is no way to form three groups and have all children satisfied.) One way to form two groups is for the child who is satisfied with one book to go alone in one group, and all the others in the other group. Then, the number of children in the largest group will be 4. Another way is for one of the children satisfied with two books to move to the first group (together with the child satisfied with one book). Then, the number of children in the largest group will be 3. It is not possible to form two groups with the largest group having fewer than 3 children, so $K = 3$.