Mock CCC '15 S2 - Problem-setting Pandemonium

Time limit: 1.0s   **Memory limit:** 64M

2015 Mock CCC by Alex and Timothy

You are planning to run N $(1 \leq N \leq 100\,000)$ programming contests in the next few weeks, but you quickly realized that you've bitten off more than you can chew. As a novice problem-setter, you quickly learned that the rumours are true. Indeed, original and interesting problems are really hard to write.

Fortunately, you're not alone in your contest-writing endeavours. Your friend is there with you. However, you know deep down that in order to be credited for helping create the contest, you just need to supply a single problem for it. You know that your friend will take care of the rest, even if it means he'll be very angry for having to do all the remaining work.

The contests are numbered from 1 to N. Both of you have agreed that in the ith $(1 \leq i \leq N)$ contest, you must contribute a problem of difficulty level D_i $(1 \leq D_i \leq 100\,000)$. You don't want to do a lot of work, so you want to create as few problems as possible to get the job done. Your master plan is to repeat problems when multiple contests require problems of the same difficulty. Surely, nobody will notice.

However, you also realized that your friend is smarter than you think. He will easily notice if you repeat the same problem for multiple contests. Your (not-so-)subtle way to circumvent this is by creating a problem and then repeatedly decreasing its difficulty (by some positive integer value) whenever you reuse it in a different contest. This way, you are less likely to be busted.

Since you will focus solely on making problems for the next few days (who needs to eat or sleep?), you are able to make an original problem of any difficulty level if you try hard enough. What is the minimum number of original problems that you'll need to write for the N contests?

Input Specification

Line 1 of input will contain a single integer N, representing the number of contests you're responsible for.
Line 2 of input will contain N space-separated integers $D_1, D_2, \ldots, D_N$, the difficulty levels of the problems you have to contribute for the contests.

Output Specification

The first and only line of output should consist of a single integer, the minimum number of original problems you have to write to be able to supply a problem for all the contests.

Sample Input 1

```
5
1 2 3 4 6
```

Sample Output 1

```
1
```

Explanation for Sample 1

Just create one original problem of difficulty 6 for contest 5, and reduce its difficulty to 4, 3, 2, and 1 in that order so you can reuse it on the other four contests.

Sample Input 2

```
6
3 2 2 1 3 3
```

Sample Output 2

```
3
```

Explanation for Sample 2

One possible solution is to write three original problems with difficulty 3.
For the first original problem: use it for contest 1, then lower it to difficulty 2 to use on contest 2, and finally to difficulty 1 to be used on contest 4.
For the second original problem: use it for contest 5, then lower it to difficulty 2 to use on contest 3.
For the last original problem, just use it for contest 6.