Problem J. J

Time limit 3000 ms
Mem limit 262144 kB

You're given an array a initially containing n integers. In one operation, you must do the following:

- Choose a position i such that $1 < i \leq |a|$ and $a_i = |a| + 1 - i$, where $|a|$ is the **current** size of the array.
- Append $i - 1$ zeros onto the end of a.

After performing this operation as many times as you want, what is the maximum possible length of the array a?

Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \leq t \leq 1000$). The description of the test cases follows.

The first line of each test case contains n ($1 \leq n \leq 3 \cdot 10^5$) — the length of the array a.

The second line of each test case contains n integers $a_1, a_2, \ldots, a_n$ ($1 \leq a_i \leq 10^{12}$).

It is guaranteed that the sum of n over all test cases does not exceed $3 \cdot 10^5$.

Output

For each test case, output a single integer — the maximum possible length of a after performing some sequence of operations.

Examples

Input	Output
4 5 2 4 6 2 5 5 5 4 4 5 1 4 6 8 2 3 1 1	10 11 10 1

Note

In the first test case, we can first choose $i = 4$, since $a_4 = 5 + 1 - 4 = 2$. After this, the array becomes $[2, 4, 6, 2, 5, 0, 0, 0]$. We can then choose $i = 3$ since $a_3 = 8 + 1 - 3 = 6$. After this, the array becomes $[2, 4, 6, 2, 5, 0, 0, 0, 0, 0]$, which has a length of 10. It can be shown that no sequence of operations will make the final array longer.

In the second test case, we can choose $i = 2$, then $i = 3$, then $i = 4$. The final array will be $[5, 4, 4, 5, 1, 0, 0, 0, 0, 0, 0]$, with a length of 11.