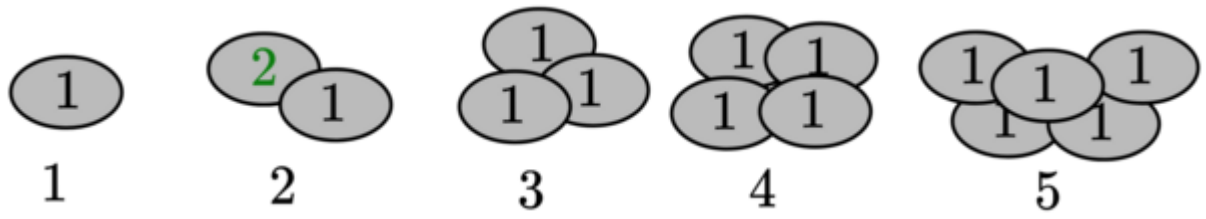
Problem E. Interview

Time limit 2000 ms
Mem limit 262144 kB

This is an interactive problem. If you are unsure how interactive problems work, then it is recommended to read the guide for participants.

Before the last stage of the exam, the director conducted an interview. He gave Gon n piles of stones, the i-th pile having a_i stones.

Each stone is identical and weighs 1 grams, except for one special stone that is part of an unknown pile and weighs 2 grams.



A picture of the first test case. Pile 2 has the special stone. The piles have weights of $1, 3, 3, 4, 5$, respectively.

Gon can only ask the director questions of one kind: he can choose k piles, and the director will tell him the total weight of the piles chosen. More formally, Gon can choose an integer k ($1 \le k \le n$) and k unique piles $p_1, p_2, \ldots, p_k$ ($1 \le p_i \le n$), and the director will return the total weight $m_{p_1} + m_{p_2} + \cdots + m_{p_k}$, where m_i denotes the weight of pile i.

Gon is tasked with finding the pile that contains the special stone. However, the director is busy. Help Gon find this pile in at most **30** queries.

Input

The input data contains several test cases. The first line contains one integer t ($1 \le t \le 1000$) — the number of test cases. The description of the test cases follows.

The first line of each test case contains a single integer n ($1 \le n \le 2 \cdot 10^5$) — the number of piles.

The second line of each test case contains n integers a_i ($1 \le a_i \le 10^4$) — the number of stones in each pile.

It is guaranteed that the sum of n over all test cases does not exceed $2 \cdot 10^5$.

After reading the input for each test case, proceed with the interaction as follows.

Interaction

You can perform the operation at most **30** times to guess the pile.

To make a guess, print a line with the following format:

- ? k p_1 p_2 p_3 ... p_{k-1} p_k ($1 \le k \le n$; $1 \le p_i \le n$; all p_i are distinct) — the indices of the piles.

After each operation, you should read a line containing a single integer x — the sum of weights of the chosen piles. (Formally, $x = m_{p_1} + m_{p_2} + \cdots + m_{p_k}$.)

When you know the index of the pile with the special stone, print one line in the following format: ! m ($1 \le m \le n$).

After that, move on to the next test case, or terminate the program if there are no more test cases remaining.

If your program performs more than 30 operations for one test case or makes an invalid query, you may receive a `Wrong Answer` verdict.

After you print a query or the answer, please remember to output the end of the line and flush the output. Otherwise, you may get `Idleness limit exceeded` or some other verdict. To do this, use the following:

- `fflush(stdout)` or `cout.flush()` in C++;
- `System.out.flush()` in Java;
- `flush(output)` in Pascal;
- `stdout.flush()` in Python;
- see the documentation for other languages.

It is additionally recommended to read the interactive problems guide for participants.

Hacks

To make a hack, use the following format.

The first line should contain a single integer t ($1 \le t \le 1000$) — the number of test cases.

The first line of each test case should contain two integers n, m ($1 \le n \le 2 \cdot 10^5$) – the number of piles and the pile with the special stone.

The second line of each test case should contain n integers a_i ($1 \le a_i \le 10^4$) — the number of stones in each pile.

Note that the interactor is **not** adaptive, meaning that the answer is known before the participant asks the queries and doesn't depend on the queries asked by the participant.

Examples

Input	Output
2 5 1 2 3 4 5 11 6 3 7 1 2 3 5 3 4 2 12 6	? 4 1 2 3 4 ? 2 2 3 ? 1 2 ! 2 ? 4 2 3 5 6 ? 2 1 4 ! 7

Note

In the first test case, the stone with weight two is located in pile 2, as shown in the picture. We perform the following interaction:

- `? 4 1 2 3 4` — ask the total weight of piles 1, 2, 3, and 4. The total weight we receive back is $1 + 3 + 3 + 4 = 11$.
- `? 2 2 3` — ask the total weight of piles 2 and 3. The total weight we receive back is $3 + 3 = 6$.
- `? 1 2` — ask the total weight of pile 2. The total weight we receive back is 3.
- `! 2` — we have figured out that pile 2 contains the special stone, so we output it and move on to the next test case.

In the second test case, the stone with weight two is located on index 7. We perform the following interaction:

- `? 4 2 3 5 6` — ask the total weight of piles 2, 3, 5, and 6. The total weight we receive back is $2 + 3 + 3 + 4 = 12$.

- `? 2 1 4` — ask the total weight of piles 1 and 4. The total weight we receive back is $1 + 5 = 6$.
- `! 7` — we have somehow figured out that pile 7 contains the special stone, so we output it and end the interaction.