D. Bit Guessing Game

time limit per test: 1 second
memory limit per test: 256 megabytes

This is an interactive problem.

Kira has a hidden positive integer n, and Hayato needs to guess it.

Initially, Kira gives Hayato the value cnt — the number of unit bits in the binary notation of n. To guess n, Hayato can only do operations of one kind: choose an integer x and subtract it from n. Note that after each operation, the number n **changes**. Kira doesn't like bad requests, so if Hayato tries to subtract a number x greater than n, he will lose to Kira. After each operation, Kira gives Hayato the updated value cnt — the number of unit bits in the binary notation of the updated value of n.

Kira doesn't have much patience, so Hayato must guess the **original** value of n after no more than 30 operations.

Since Hayato is in elementary school, he asks for your help. Write a program that guesses the number n. Kira is an honest person, so he chooses the initial number n before all operations and **does not** change it afterward.

Input

The input data contains several test cases. The first line contains one integer t ($1 \le t \le 500$) — the number of test cases. The description of the test cases follows.

The first line of each test case contains the number cnt — the initial number of unit bits in the binary notation n.

The hidden integer n satisfies the following constraint: $1 \le n \le 10^9$.

Interaction

To guess n, you can perform the operation at most 30 times. To do that, print a line with the following format: "`- x`" ($1 \le x \le 10^9$).

After this operation, the number x is subtracted from n, and therefore n **is changed**. If the number x is greater than the current value of n, then the request is considered invalid.

After the operation read a line containing a single non-negative integer cnt — the number of unit bits in the binary notation of the current n after the operation.

When you know the **initial** value of n, print one line in the following format: "`! n`" ($1 \le n \le 10^9$).

After that, move on to the next test case, or terminate the program if there are none.

If your program performs more than 30 operations for one test case, subtracts a number x greater than n, or makes an incorrect request, then **response to the request will be** `-1`, after receiving such response, your program must exit immediately to receive the `Wrong Answer` verdict. Otherwise, you can get any other verdict.

After printing a query or the answer, do not forget to output the end of line and flush the output. Otherwise, you will get `Idleness limit exceeded`. To do this, use:

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

Hacks

To make a hack, use the following format.

The first line should contain a single integer t ($1 \le t \le 500$).

Each test case should contain one integer n ($1 \le n \le 10^9$) on a separate line.

Example

input Copy

```
3
1
0
1
1
0
2
1
0
```

Codeforces Round 846 (Div. 2)
Finished
Practice

→ Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

Start virtual contest

→ Clone Contest to Mashup

You can clone this contest to a mashup.

Clone Contest

→ Submit?

Language: GNU G++20 13.2 (64 bit, win

Choose file: Choose File No file chosen

Submit

→ Last submissions

Submission	Time	Verdict
278387653	Aug/27/2024 16:03	Accepted
278387514	Aug/27/2024 16:02	Accepted
278375195	Aug/27/2024 14:24	Accepted
278375039	Aug/27/2024 14:22	Wrong answer on test 3
278374910	Aug/27/2024 14:21	Wrong answer on test 1

→ Problem tags

binary search bitmasks constructive algorithms interactive *1800

No tag edit access

→ Contest materials

- Announcement
- Tutorial (en)

output Copy

```
- 1

! 1

- 1

- 1

! 2

- 2

- 1

! 3
```

Note

For example, the number of unit bits in number 6 is 2, because binary notation of 6 is 110. For 13 the number of unit bits is 3, because $13_{10} = 1101_2$.

In the first test case, $n = 1$, so the input is the number 1. After subtracting one from n, it becomes zero, so the number of unit bits in it is 0.

In the third test case, $n = 3$, which in binary representation looks like $3_{10} = 11_2$, so the input is the number of ones, that is 2. After subtracting 2, $n = 1$, so the number of unit bits is now 1. After subtracting one from n, it becomes equal to zero.

Note that the blank lines in the input and output examples are shown for clarity and are not present in the actual interaction.

Codeforces (c) Copyright 2010-2024 Mike Mirzayanov
The only programming contests Web 2.0 platform
Server time: Aug/28/2024 01:16:34 (h2).
Desktop version, switch to mobile version.
Privacy Policy

Supported by