

D. Mahmoud and Ehab and the binary string

time limit per test: 2 seconds

memory limit per test: 256 megabytes

input: standard input

output: standard output

Mahmoud and Ehab are in the fourth stage now.

Dr. Evil has a hidden binary string of length n . He guarantees that there is at least one '0' symbol and at least one '1' symbol in it. Now he wants Mahmoud and Ehab to find a position of any '0' symbol and any '1' symbol. In order to do this, Mahmoud and Ehab can ask Dr. Evil up to 15 questions. They tell Dr. Evil some binary string of length n , and Dr. Evil tells the Hamming distance between these two strings. Hamming distance between 2 binary strings of the same length is the number of positions in which they have different symbols. You can find the definition of Hamming distance in the notes section below.

Help Mahmoud and Ehab find these two positions.

You will get *Wrong Answer* verdict if

- Your queries doesn't satisfy interaction protocol described below.
- You ask strictly more than 15 questions and your program terminated after exceeding queries limit. Please note, that you can do up to 15 ask queries and one answer query.
- Your final answer is not correct.

You will get *Idleness Limit Exceeded* if you don't print anything or if you forget to flush the output, including for the final answer (more info about flushing output below).

If you exceed the maximum number of queries, You should terminate with 0, In this case you'll get Wrong Answer, If you don't terminate you may receive any verdict because you'll be reading from a closed stream .

Input

The first line of input will contain a single integer n ($2 \leq n \leq 1000$) — the length of the hidden binary string.

Output

To print the final answer, print "! pos0 pos1" (without quotes), where $pos0$ and $pos1$ are positions of some '0' and some '1' in the string (the string is 1-indexed). **Don't forget to flush the output after printing the answer!**

Interaction

To ask a question use the format "? s" (without quotes), where s is a query string. **Don't forget to flush the output after printing a query!**

After each query you can read a single integer from standard input — the Hamming distance between the hidden string and the query string.

To flush the output you can use:-

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

Hacking.

To hack someone just print one binary string with length up to 1000, containing at least one '0' and at least one '1'.

Example

input
3 2 1 3 2 1 0
output
? 000 ? 001 ? 010 ? 011 ? 100 ? 101 ! 2 1

Note

Hamming distance definition: https://en.wikipedia.org/wiki/Hamming_distance

In the first test case the hidden binary string is **101**, The first query is **000**, so the Hamming distance is **2**. In the second query the hidden string is still **101** and query is **001**, so the Hamming distance is **1**.

After some queries you find that symbol at position **2** is '0' and symbol at position **1** is '1', so you print "**! 2 1**".