Codeforces Round #371 (Div. 2)

A. Meeting of Old Friends

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Today an outstanding event is going to happen in the forest — hedgehog Filya will come to his old fried Sonya!

Sonya is an owl and she sleeps during the day and stay awake from minute l_1 to minute r_1 inclusive. Also, during the minute k she prinks and is unavailable for Filya.

Filya works a lot and he plans to visit Sonya from minute l_2 to minute r_2 inclusive.

Calculate the number of minutes they will be able to spend together.

Input

The only line of the input contains integers l_1, r_1, l_2, r_2 and k ($1 \le l_1, r_1, l_2, r_2, k \le 10^{18}$, $l_1 \le r_1$, $l_2 \le r_2$), providing the segments of time for Sonya and Filya and the moment of time when Sonya prinks.

Output

Print one integer — the number of minutes Sonya and Filya will be able to spend together.

Examples

input
1 10 9 20 1
output
2

input
1 100 50 200 75
output
50

Note

In the first sample, they will be together during minutes 9 and 10.

In the second sample, they will be together from minute 50 to minute 74 and from minute 76 to minute 100.

B. Filya and Homework

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Today, hedgehog Filya went to school for the very first time! Teacher gave him a homework which Filya was unable to complete without your help.

Filya is given an array of non-negative integers $a_1, a_2, ..., a_n$. First, he pick an integer x and then he adds x to some elements of the array (no more than once), subtract x from some other elements (also, no more than once) and do no change other elements. He wants all elements of the array to be equal.

Now he wonders if it's possible to pick such integer x and change some elements of the array using this x in order to make all elements equal.

Input

The first line of the input contains an integer n ($1 \le n \le 100\,000$) — the number of integers in the Filya's array. The second line contains n integers $a_1, a_2, ..., a_n$ ($0 \le a_i \le 10^9$) — elements of the array.

Output

If it's impossible to make all elements of the array equal using the process given in the problem statement, then print "NO" (without quotes) in the only line of the output. Otherwise print "YES" (without quotes).

Examples

input
5 1 3 3 2 1
output
YES

input
5 1 2 3 4 5
output
NO

Note

In the first sample Filya should select $x = 1$, then add it to the first and the last elements of the array and subtract from the second and the third elements.

C. Sonya and Queries

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Today Sonya learned about long integers and invited all her friends to share the fun. Sonya has an initially empty multiset with integers. Friends give her t queries, each of one of the following type:

1. $+ \ a_i$ — add non-negative integer a_i to the multiset. Note, that she has a multiset, thus there may be many occurrences of the same integer.
2. $- \ a_i$ — delete a single occurrence of non-negative integer a_i from the multiset. It's guaranteed, that there is at least one a_i in the multiset.
3. $? \ s$ — count the number of integers in the multiset (with repetitions) that match some pattern s consisting of 0 and 1. In the pattern, 0 stands for the even digits, while 1 stands for the odd. Integer x matches the pattern s, if the parity of the i-th from the right digit in decimal notation matches the i-th from the right digit of the pattern. If the pattern is shorter than this integer, it's supplemented with 0-s from the left. Similarly, if the integer is shorter than the pattern its decimal notation is supplemented with the 0-s from the left.

For example, if the pattern is $s = 010$, than integers $92, 2212, 50$ and 414 match the pattern, while integers $3, 110, 25$ and 1030 do not.

Input

The first line of the input contains an integer t ($1 \le t \le 100\,000$) — the number of operation Sonya has to perform.

Next t lines provide the descriptions of the queries in order they appear in the input file. The i-th row starts with a character c_i — the type of the corresponding operation. If c_i is equal to '+' or '−' then it's followed by a space and an integer a_i ($0 \le a_i < 10^{18}$) given without leading zeroes (unless it's 0). If c_i equals '?' then it's followed by a space and a sequence of zeroes and onse, giving the pattern of length no more than 18.

It's guaranteed that there will be at least one query of type '?'.

It's guaranteed that any time some integer is removed from the multiset, there will be at least one occurrence of this integer in it.

Output

For each query of the third type print the number of integers matching the given pattern. Each integer is counted as many times, as it appears in the multiset at this moment of time.

Examples

input
12 + 1 + 241 ? 1 + 361 - 241 ? 0101 + 101 ? 101 - 101 ? 101 + 4000 ? 0
output
2 1 2 1 1

input
4 + 200 + 200 - 200 ? 0
output
1

Note

Consider the integers matching the patterns from the queries of the third type. Queries are numbered in the order they appear in the input.

1. 1 and 241.
2. 361.
3. 101 and 361.
4. 361.
5. 4000.

D. Searching Rectangles

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Filya just learned new geometry object — rectangle. He is given a field consisting of $n \times n$ unit cells. Rows are numbered from bottom to top with integer from 1 to n. Columns are numbered from left to right with integers from 1 to n. Cell, located at the intersection of the row r and column c is denoted as (r, c). Filya has painted two rectangles, such that their sides are parallel to coordinate axes and each cell lies fully inside or fully outside each of them. Moreover, no cell lies in both rectangles.

Later, hedgehog Filya became interested in the location of his rectangles but was unable to find the sheet of paper they were painted on. They were taken by Sonya and now she wants to play a little game with Filya. He tells her a query rectangle and she replies with the number of initial rectangles that lie **fully inside** the given query rectangle. The query rectangle should match the same conditions as initial rectangles. Rectangle lies fully inside the query if each o its cells lies inside the query.

Filya knows Sonya really well, so is sure that if he asks more than 200 questions she will stop to reply.

Input

The first line of the input contains an integer n ($2 \le n \le 2^{16}$) — size of the field.

For each query an integer between 0 and 2 is returned — the number of initial rectangles that lie fully inside the query rectangle.

Output

To make a query you have to print "`?` x_1 y_1 x_2 y_2" (without quotes) ($1 \le x_1 \le x_2 \le n$, $1 \le y_1 \le y_2 \le n$), where (x_1, y_1) stands for the position of the bottom left cell of the query and (x_2, y_2) stands for the up right cell of the query. You are allowed to ask no more than 200 queries. After each query you should perform "flush" operation and read the answer.

In case you suppose you've already determined the location of two rectangles (or run out of queries) you should print "`!` x_{11} y_{11} x_{12} y_{12} x_{21} y_{21} x_{22} y_{22}" (without quotes), where first four integers describe the bottom left and up right cells of the first rectangle, and following four describe the corresponding cells of the second rectangle. You can print the rectangles in an arbitrary order. After you have printed the answer, print the end of the line and perform "flush". Your program should terminate immediately after it print the answer.

Interaction

To flush you can use (just after printing an integer and end-of-line):

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

You will get the `Wrong Answer` verdict if you ask more than 200 queries, or if you print an incorrect coordinates.

You will get the `Idleness Limit Exceeded` verdict if you don't print anything (but you should) or if you forget about flushing the output (more info below).

Hacking.

The first line should contain an integer n ($2 \le n \le 2^{16}$).

The second line should contain four integers x_1, y_1, x_2, y_2 ($1 \le x_1 \le x_2 \le n$, $1 \le y_1 \le y_2 \le n$) — the description of the first rectangle.

The third line contains the description of the second rectangle in the similar way.

Example

input
5 2 1 0 1 1 1 0 1
output
? 1 1 5 5 ? 1 1 3 3 ? 1 1 3 1 ? 2 2 2 2 ? 3 3 5 5 ? 3 3 3 5 ? 3 3 3 4 ? 3 4 3 5 ! 2 2 2 2 3 4 3 5

E. Sonya and Problem Wihtout a Legend

time limit per test: 5 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Sonya was unable to think of a story for this problem, so here comes the formal description.

You are given the array containing n positive integers. At one turn you can pick any element and increase or decrease it by 1. The goal is the make the array strictly increasing by making the minimum possible number of operations. You are allowed to change elements in any way, they can become negative or equal to 0.

Input

The first line of the input contains a single integer n $(1 \leq n \leq 3000)$ — the length of the array.

Next line contains n integer a_i $(1 \leq a_i \leq 10^9)$.

Output

Print the minimum number of operation required to make the array strictly increasing.

Examples

input
7 2 1 5 11 5 9 11
output
9

input
5 5 4 3 2 1
output
12

Note

In the first sample, the array is going to look as follows:

$2\ 3\ 5\ 6\ 7\ 9\ 11$

$|2 - 2| + |1 - 3| + |5 - 5| + |11 - 6| + |5 - 7| + |9 - 9| + |11 - 11| = 9$

And for the second sample:

$1\ 2\ 3\ 4\ 5$

$|5 - 1| + |4 - 2| + |3 - 3| + |2 - 4| + |1 - 5| = 12$