Problem A. Autonomous Cities

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	512 mebibytes

There are N cities in Byteland, cities are connected by $N-1$ bidirectional roads such as each city can be reached from the another one using one of several those roads. A city is called *autonomous*, if it have only one road leading to (and from) this city.

Given N, find the least possible number of autonomous cities.

Input

Input consists of one integer N ($2 \le N \le 10^9$) — number of cities in Byteland.

Output

Print one integer — least possible number of autonomous cities.

Examples

standard input	standard output
2	2
3	2

Problem B. Binary Operations

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	512 mebibytes

This is an interactive problem.

The jury secretly choose one of the following binary operations: '`+`' (addition), '`-`' (**absolute value** of difference), '`*`' (multiplication), '`/`' (integer division), '`|`' (bitwise OR), '`&`' (bitwise AND) or '`^`' (bitwise XOR).

You must ask **only one** query in form "`? X Y`", where X and Y are digits (i.e. integers between '`0`' and '`9`' inclusively). Then the jury program will apply the secret operation to your integers and returns the answer as non-negative integer. Note that after '`-`' the result $|X - Y|$ is printed, i.e. answer will be non-negative.

Immediately after that you shall name the secret operation.

Interaction Protocol

The interaction starts with query from your program in form "`?` X Y", where X and Y are integers between 0 and 9 inclusively.

If the query is well-formed, you will receive the result of applying of the secret operation to X and Y.

Then you shall print the answer.

- "`! +`" for the addition.
- "`! -`" for the subtraction.
- "`! *`" for the multiplication.
- "`! /`" for the division.
- "`! |`" for bitwise AND.
- "`! &`" for bitwise OR.
- "`! ^`" for bitwise XOR.

Dont forget to print after each of two your actions exactly one newline character and flush the output buffer using the "`flush`" function of your programming language. Otherwise your solution will receive "Idleness Limit Exceeded" verdict.

Example

standard input	standard output
42	? 6 7 !*

Problem C. Customity

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	512 mebibytes

Let's define *customity* of two adjacent elements a_i and a_{i+1} of the array A as $|a_i - a_{i+1}|$.

Given an integer array A. For one operation you may add to any element of the sequence arbitrary real number. Your task is to convert the array to one where maximum customity will be minimized.

Calculate minimal number of operations needed to do that.

Input

First line of the input contains one integer N — length of the given array ($2 \le N \le 10^5$). Second line contains N integers a_i ($-10^6 \le a_i \le 10^6$) — elements of the given array A.

Output

Print one integer — minimal number of operations needed to obtain the array where maximum customity is minimized.

Examples

standard input	standard output
2 1083 6006	1

Problem D. Divide and Convert

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	512 mebibytes

Given a string, consisting of letters 'I', 'V', 'X', 'L', 'C', 'D' and 'M'. Your task is to split the string to the continuous substrings such as:

- Each substring is a valid Roman numeral.
- The sum of integers, represented by those numerals, is maximum possible.

Input

Input contains one non-empty string, consisting of letters 'I', 'V', 'X', 'L', 'C', 'D' and 'M'. Length of the string does not exceed 10^6.

Output

Print one integer — maximal sum of the integers in Roman system, achieved by splitting the string to the valid Roman numerals.

Examples

standard input	standard output
VII	7
ICL	151

Note

In the first sample we can take the number as it is. In the second sample the maximum can be reached, for example, by splitting the string by two substrings: 'I' (1) and 'CL' (150), what gives 151 as the sum.

Problem E. Establish The Minimum

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	512 mebibytes

Given an number. Reorder the characters in its decimal representation in a way such as the resulting integer will be minimal possible.

Both given and resulting number shall conform the following rules to be valid:

- Decimal representation consists of digits between '0' and '9' inclusively and special characters '-' and '.', usage of those characters is described later. You may assume that representation contains atleast one digit.
- The number may be negative (then first character is representation is the '-') or non-negative (then no special character is used). There cannot be '-' character anywhere else than at first position.
- The number may be finite decimal fraction (in this case the representation contains exactly one dot '.' as decimal point; the dot can be placed only between two digits) or integer, in this case decimal representation does not contains '.'.
- The representation may start from zero (or have zero as second character after '-' sign) only if the number is equal to zero or if immediately after it goes the decimal point. However, zeroes at the end of the representation of fractional numbers are OK, for example, the representations -123.200, or 2.0000, or 0.0 are valid.
- If the absolute value of the given integer is equal to zero, the '-' sign cannot be used.
- Lentgh of the decimal representation is not greater than 10^4.

Input

Input file contains one non-empty string consisting of no more than 10^4 characters — valid decimal representation of the number.

Output

Print minimal possible valid number which can be obtained by rearranging characters in the given decimal representation.

Examples

standard input	standard output
123405	102345
-3.1416036	-6643311.0

Problem F. Find Two Triangles

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	512 mebibytes

Find two triangles with integer lengths of sides, given perimeter P, one with maximal non-zero area, other with minimal non-zero area.

Input

Input contains one integer P — given value of the perimeter.

Output

In the first line print three integers — lengths of sides of the triangle with given perimeter and **maximal** area. In the second line print in same format lengths of sides of the triangle with given perimeter and **minimal non-zero** area. If there is more than one solution, print any of them.

If there are no non-degenerated triangles with integer lengths of sides and perimeter P, print -1 instead.

Examples

standard input	standard output
3	1 1 1 1 1 1
4	-1

Problem G. Guess The Integer

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	512 mebibytes

This is an interactive problem.

Your task is to guess some secret positive integer N.

Initially, the value of the variable v in the memory of the jury program is N.

Each time jury program tells you the sum of digits if decimal representation of the current value of v. For one query you may ask the jury program to decrease v by k, where k is positive integer. If the difference become negative, you fail immediately. Otherwise jury program decreases v by k and tells you sum of digits of the new value of v.

When your program have enough information to guess N, it shall tell the value of N to the jury program. If your guess is right, you never failed and used no more than 128 queries (note that naming the answer is not counted for query), then your solution is passed the test case.

Interaction Protocol

At the beginning of the interaction the jury program prints one integer — sum of digits in the decimal representation of the integer N (and sets $v = N$).

Then you may do the queries. Each query have the form "`?` k", where k is positive integer, does not exceeding 10^{18}. It means that v shall be decreased by k.

If the query have wrong format, or $v - k$ become negative, or there are more than 128 queries used, interaction immediately stops and the test considered failed. Otherwise value of v is decreased by k, and jury program tells you sum of digits in the decimal representation of the new value of v.

If you want to print the answer, print "`!` X", where X is guessed value of N. You may assume that N does not exceeds 10^{18} and that jury program is not adaptive (i.e. never changes N in the process of guessing). This action does not counted as query.

Dont forget to print after each of your actions exactly one newline character and flush the output buffer using the "`flush`" function of your programming language. Otherwise your solution will receive "Idleness Limit Exceeded" verdict.

Example

standard input	standard output
3 2 0	 ? 1 ? 11 ! 12

Note

Note that in case $N = 3$ query "? 11" cause immediate fail.

Problem H. Hunt For 2020's!

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	512 mebibytes

Hassan got a string S $(s_1 \dots s_n)$ of length n consisting of only digits '0', '1', and '2', and he wants to pick some **disjoint** subsequences which equal to 2020, as many as possible.

Formally, Hassan would like to find k quadruples $(a_1, b_1, c_1, d_1), \dots, (a_k, b_k, c_k, d_k)$ such as

- $1 \le a_i < b_i < c_i < d_i \le n$
- $s_{a_i} s_{b_i} s_{c_i} s_{d_i} = 2020$
- $\{a_i, b_i, c_i, d_i\} \cap \{a_j, b_j, c_j, d_j\} = \emptyset$ for $i \ne j$.

Find the maximum value of k.

Input

The input consists of several test cases terminated by end-of-file.

The first line of each test case contains an integer n $(1 \le n \le 10^5)$. Second line contains the string S $(s_1 \dots s_n)$. $(s_i \in \{0, 1, 2\})$. Sum of n in all test cases does not exceed 10^6.

Output

For each test case print an integer which denotes the result.

Examples

standard input	standard output
4 2222 7 2101210 9 122002200	0 1 2

Problem I. Implement Efficiently!

Input file:	*standard input*
Output file:	*standard output*
Time limit:	5 seconds
Memory limit:	512 mebibytes

Iveri has two sequences $a_1, a_2, \ldots, a_n$ and $b_1, b_2, \ldots, b_n$. He would like to perform the following operations:

- `1` x y, change the value of a_x to y.
- `2` x y, change the value of b_x to y.
- `3` x, find the value of c_x, where $c_0 = 0$, $c_i = \max(c_{i-1} + b_i, a_i)$ for $1 \le i \le x$.

Implement an efficient data structure to process those operations.

Input

The input consists of several test cases terminated by end-of-file. For each test case:

The first line contains two integers n and m , which are the length of the two sequences and the number of operations. The second line contains n integers $a_1, a_2, \ldots, a_n$. The third line contains n integers $b_1, b_2, \ldots, b_n$. Each of the last m lines contains a query.

- $1 \le n, m \le 2 \times 10^5$
- $-10^9 \le a_i, b_i, y \le 10^9$
- $1 \le x \le n$
- The sum of n and the sum of m do not exceed 2×10^6.

Output

For each query of type 3, output an integer denoting the value of c_x.

Example

standard input	standard output
4 9 1 2 3 3 -1 2 3 3 3 1 3 2 3 3 3 4 2 2 -4 3 1 3 2 3 3 3 4	1 3 6 9 1 2 5 8

Problem J. Just Count Them

Input file:	*standard input*
Output file:	*standard output*
Time limit:	5 seconds
Memory limit:	512 mebibytes

Jamal has a sequence $a_1, a_2, \ldots, a_n$. He can rearrange the sequence using the following operation any number of times:

- Select an integer i $(1 \le i \le n)$ and change the sequence to $a_i, a_{i-1}, \ldots, a_1, a_n, a_{n-1}, \ldots, a_{i+1}$.

Jamal would like to know the number of different sequences can be obtained modulo $(10^9 + 7)$.

Input

The input consists of several test cases terminated by end-of-file. For each test case:

The first line contains an integer n, the length of the sequence.

The second line contains n integers $a_1, a_2, \ldots, a_n$.

- $1 \le n \le 10^5$
- $1 \le a_i \le n$
- The sum of n does not exceed 2×10^6.

Output

For each test case, print an integer which denotes the result.

Example

standard input	standard output
4 1 1 1 1 4 1 1 2 2 4 1 2 1 2 4 2 1 2 1	1 4 2 2

Problem K. Kingdom of Onliners

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	512 mebibytes

In Kingdom of Onliners N popular bloggers are publishing the messages.

Messages of other bloggers are reposted in the following way: given list of **directed** pairs of blogers (u, v) such as bloger v reposts anything bloger u posted (but reverse may not be true).

You are responsible for the PR-campaign of some brand new smartphone in Kingdom of Onliners, and came with brilliant plan to find the blogger, whose messages will be reposted by all bloggers, and persuade her to post something good about your product. Lets call such a blogger "influencer".

So, given all the information about directions of reposting, before choosing an influencer you are decided to make full list of them.

Input

The first line of the input contains two integers N a M: the number of bloggers and the number of reposting directions.

Then, M more lines follow. Each contains two integers A_i and B_i and denotes that bloger B_i is reposting messages of A_i.

It holds $1 \le N, M \le 10^6$.
For all i it holds $1 \le A_i, B_i \le N$ and $A_i \ne B_i$.

No two directed pair repeats (but there may be pair of bloggers, reposting each other).

Output

Output two lines. The first line should contain the total number of influencers. The second line should contain a space-separated list of these infiuencers **in ascending order**.

Example

standard input	standard output
5 6 2 3 2 5 3 1 3 4 4 2 5 4	4 2 3 4 5
3 2 2 1 3 1	0

Problem L. Longest Subsequences

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	512 mebibytes

Luka has two strings s and t. He would like to choose two subsequences x from s and y from t such that:

- x is lexicographically smaller than or equal to y.
- The sum of $|x|$ and $|y|$ is maximal, where $|s|$ denotes the length of the string s.

Note that:

- Both x and y could be empty string.
- A subsequence is a sequence that can be derived from the given sequence by deleting zero or more elements without changing the order of the remaining elements.
- String x is lexicographically less than string y, if either x is a prefix of y (and $x \neq y$), or there exists such i $(1 \leq i \leq \min(|x|, |y|))$, that $x_i < y_i$, and for any j $(1 \leq j < i)$ $x_j = y_j$.

Input

The input consists of several test cases terminated by end-of-file. For each test case:

The first line contains a string s. The second line contains a string t.

- $1 \leq |s| \leq 2000$
- $1 \leq |t| \leq 2000$
- The sum of $|s|$ does not exceed 20000.
- The sum of $|t|$ does not exceed 20000.
- Both the strings consist only of English lowercase letters.

Output

For each test case, output the sum of $|x|$ and $|y|$.

Example

standard input	standard output
aaaa bbbb abcd abca abcd abcd	8 7 8

Problem M. Matrix

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	512 mebibytes

Mariam has an integer x and two n arrays $a_1, \ldots, a_n$, $b_1, \ldots, b_n$. She makes an $n \times n$ matrix M where

$$M_{i,j} = \begin{cases} x + a_i b_j & \text{when } i = j \\ a_i b_j & \text{otherwise} \end{cases}$$

Find the determinant of the matrix M modulo $(10^9 + 7)$.

Input

The input consists of several test cases terminated by end-of-file.

The first line of each test case contains two integers n and x. The second line contains n integers $a_1, \ldots, a_n$. The third line contains n integers $b_1, \ldots, b_n$.

- $1 \le n \le 10^5$
- $0 \le x, a_i, b_i \le 10^9$
- The sum of n does not exceed 10^6.

Output

For each test case, print an integer which denotes the result.

Examples

standard input	standard output
2 1 0 0 0 0 2 1 1000000000 1000000000 1000000000 1000000000 3 2 2 3 3 2 3 3	1 99 96