Problem A. Approach

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

Let's consider The National University of Colombia (UNAL) as a large coordinate plane centered at the most emblematic place of the University, the Che Square.
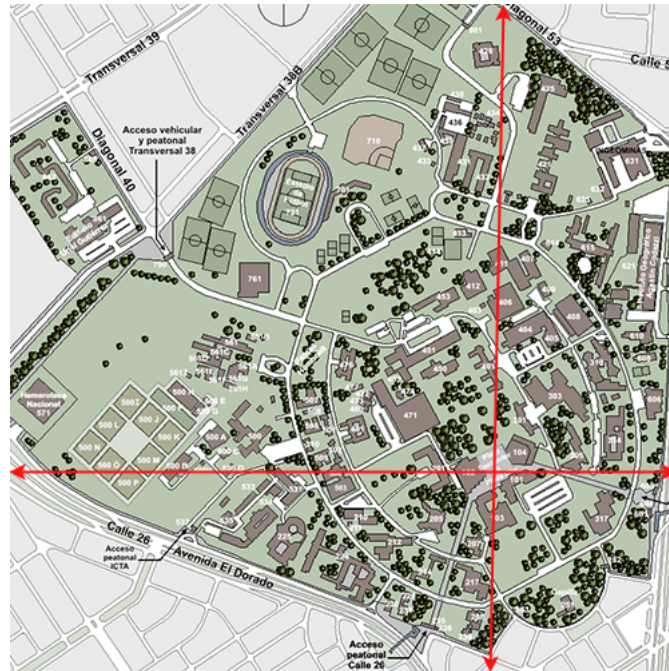
Two friends are attending classes and now they may need to move in order to attend their next classes. The first of them is currently at a building located at A and needs to go to the building located at B, the second friend is currently at a building located at C and needs to go to the building located at D.

They start to move at the same time, both of them move in a straight line and with the same constant speed. If one of them reaches his destination first, he will remain there, while the other one is still moving.

Your task is to find the minimum distance between the two friends during their change of classes.

Input

You are given two lines each of them containing 4 integers separated by spaces. The first line contains $A.x$, $A.y$, $B.x$ and $B.y$. The second line contains $C.x$, $C.y$, $D.x$ and $D.y$. The absolute value of each coordinate doesn't exceed 10^9.

Output

Output one line containing the answer to the task. Your answer will be considered correct if the absolute or relative error is at most 10^{-6}.

Examples

standard input	standard output
0 0 1 0 2 0 2 1	1.414213562373
-2 -4 -2 -2 -3 -5 -3 -2	1.000000000000

Problem B. Baby name

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

A cute couple is gonna have a new baby soon, they don't know yet how they will name their baby. However, they want the name of the baby to be the concatenation of a non-empty substring of the father's name and a non-empty substring of the mother's name. Formally, let $a = a_1 a_2 ... a_n$ be the name of the father and $b = b_1 b_2 ... b_m$ be the name of the mother, the name of the baby will be $a_i a_{i+1} ... a_{i+s} b_j b_{j+1} ... b_{j+t}$, where $1 \le i \le i + s \le n$ and $1 \le j \le j + t \le m$.

Both the mother and the father, are UNAL alumni. They know that having a name at the end of the course list in alphabetic order is very advantageous because that person is always the last being called, which means having more time to arrive to the class, to deliver homework, to prepare presentations, and so on.

To determine which of the two names is higher alphabetically, their first letters are compared. If they differ, then the name whose first letter comes later in the alphabet is higher than the other name. If the first letters are the same, then the second letters are compared, and so on. If a position is reached where one name has no more letters to compare while the other does, then the largest name is the higher alphabetically.

For example, *andres* is higher alphabetically than *ana*, *andre* and *andrea*, but it is lower than *cadie*, *andy* and *andresito*.

The couple knows that in the future the baby will study at UNAL, and of course, they want to help the baby to do very well. Considering this, they want the name of the baby to be the highest alphabetically possible.

Given the name of the father and the name of the mother, what should be the name for the baby?

Input

The input consist of 2 lines, first line contains the name of the father and the second name contains the name of the mother.

Each name is a non-empty string consisting of at most $2 * 10^5$ lowercase english characters.

Output

Print one line with the name of the baby.

Examples

standard input	standard output
jose maria	sria
abel sun	lun

Problem C. Cipher count

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	512 megabytes

The Vigenère cipher is a very famous method for encrypting messages using polyalphabetic substitution, this method is a generalization of Cesar's cipher or ROT cipher, which do shifts of the letter over an alphabet. Formally, Vigenère works as follows:

Let Σ be the alphabet of size a, which can be mapped to numbers $0, 1, ..., a - 1$. let $m = m_1 m_2 ... m_n$ be a message of size n, using the letters from Σ, and $k = k_1 k_2 ... k_t$ the key of size t, using also letters from Σ. To encrypt, we concatenate k to itself multiple times until it has size n, then we shift each letter in m some positions to the right in the alphabet according to the letter of the repeated key, i.e. for each letter m_i we take the letter in the alphabet that is k_i positions to the right cyclically. To decrypt, we do the same procedure but shifting the letters to the left instead.

For example, take the 26 letters of the English alphabet from A to Z, where A is mapped to 0, B is mapped to 1, and so on, Z is mapped to 25. We want to encrypt the message $m = $ MESSAGE of size 7 using the key $k = $ KEY of size 3, first we concatenate k multiple times until its size is 7, obtaining KEYKEYK. Then we shift each letter in m in the alphabet according to the repeated key. For instance, the first M in the message is shifted K positions (10 positions) in the alphabet mapping it to the letter W. Note that the shifting is cyclic, therefore after letter Z goes letter A. The whole encryption process look like this:

$$m = \text{MESSAGE}$$
$$\mid\mid\mid\mid\mid\mid\mid$$
$$k = \text{KEYKEYK}$$
$$\mid\mid\mid\mid\mid\mid\mid$$
$$e = \text{WIQCEEO}$$

Note that Vigenère cipher, as most symmetric ciphers, depends only on the key. Thus, if an attacker is able to get the correct key, the attacker could read every encrypted juicy message.

Now, an attacker knows the size of the alphabet a and the maximum size of the key. What is the minimum number of keys the attacker has to try, to be sure to get a key which can decrypt any message sent by the sender?

Input

The input consists of 2 numbers a $(1 \le a \le 10^3)$ and k $(1 \le k \le 5 * 10^6)$ — The size of the alphabet and the maximum size of the key, respectively.

Output

Print one single number, the number of keys the attacker should try to be sure he (or she) can decrypt correctly any message.

As this number can be very large print it modulo $10^9 + 7$.

Examples

standard input	standard output
`26 1`	`26`
`2 2`	`4`
`1 3`	`1`

Note

For the first sample, consider the English alphabet of size a = 26 and a maximum key size k = 1. The attacker has to try 26 possible keys: a, b, c,..., z.

For the second sample, consider the binary alphabet 0, 1 of size a = 2 and a maximum key size k = 2. The attacker can decrypt any message after trying only 4 keys: 0, 1, 01 and 10. Note that it is not necessary to try key 00, as any message encrypted with key 00 can be decrypted with key 0, it is not necessary to try key 11 neither.

Problem D. Dices

Input file:	`standard input`
Output file:	`standard output`
Time limit:	6 seconds
Memory limit:	512 megabytes

Diego has bought n dices to play with, all dices have k sides numbered $1, 2, ..., k$, also as Diego wants to trick his friends, all dices are loaded, meaning that the probability of rolling the dice is not the same for all sides.

In particular, Diego loaded the dice so that all sides which are multiple of m have zero probability of appearance, while all other sides have the equal probability of appearance. Diego thinks that this will be enough to win any game betting against multiples of m. Ivan, a friend of Diego, knows the dices are loaded and he thinks that after summing the values of the dices we can still get a good probability of having a multiple of m.

For example, if Diego has 2 dices with 6 sides loaded, such that multiples of 3 cannot appear, then each dice could roll into 1, 2, 4 or 5 with probability 1/4 each. After throwing both dices Diego has the following possible outcomes for multiples of 3:

- 3 with 1/8 of probability

- 6 with 1/4 of probability

- 9 with 1/8 of probability

Therefore, in this case, there is a chance of 1/2 of having a multiple of 3, even if each dice, individually, has 0 chance of having a multiple of 3.

Help Ivan to calculate the probability of having a multiple of m after summing the result of rolling all n loaded dices.

Input

The input consists of 3 integers n ($1 \leq n \leq 2 * 10^6$), k ($2 \leq k \leq 2 * 10^6$) and m ($2 \leq m \leq 200$) — The number of dices, the number of sides in each dice and the multiple chosen to load the dices, respectively

Output

Let p/q be the probability of having a multiple of m after rolling the dices. Print just a single integer: $p * q^{-1} \mod 1000000007$.

Examples

standard input	standard output
2 2 2	1
3 2 2	0
2 6 3	500000004

Problem E. Enter to the best problem of this contest!

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

This is an interactive problem. *You have to use a `flush` operation right after printing each line. For example, in C++ you should use the function `fflush(stdout)` or `cout.flush()`, in Java — `System.out.flush()`, in Pascal — `flush(output)` and in Python — `sys.stdout.flush()`.*

Osman is lost again :(

He was trying to come up with the best problem of this contest so he didn't notice that he was entering a maze. Oh, poor Osman!

Osman is lost in a really symmetrical maze: The maze looks like a perfect binary tree of depth n ($1 \leq n \leq 29$) and, luckily, we have a map of the maze. Each node of the binary tree is named by a number and, for each node k, the two children of this node (if any) are named by $2k$ and $2k + 1$. For instance, when the depth of the binary tree is 4, then the maze looks like this:
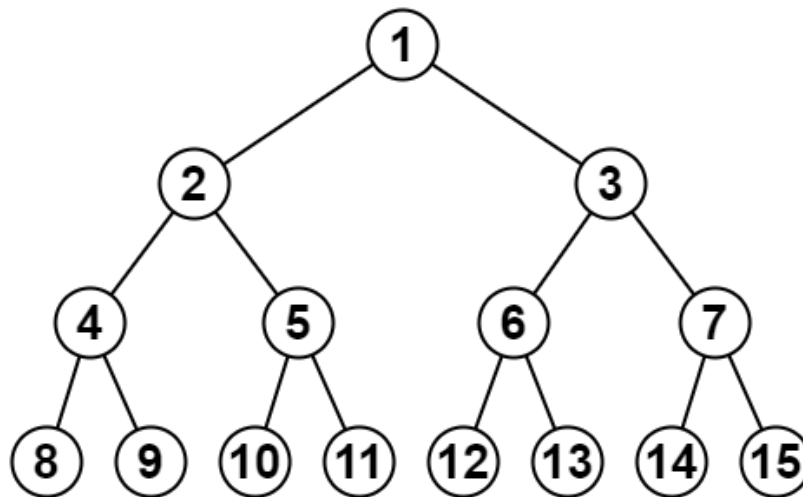
Your task is to find Osman. He is unable to move in one of the nodes of the tree. You have a device where you can make queries, each query is a single integer k from 1 to $2^n - 1$. Flush the output stream after printing each query. The device will provide you with the distance from k to the node where Osman is. You can do at most 29 queries to the device before it explodes (or maybe it doesn't, but you don't want to know, do you?).

Once you find Osman, print "! x" (without quotes), where x is the node where Osman is, and terminate your program normally immediately after flushing the output stream.

Help Osman exit the maze! Or not, I don't care.

Input

Use standard input to read the responses to the queries.

The first line contains an integer n ($1 \leq n \leq 29$) — The depth of the tree.

Following lines will contain responses to your queries — The distance between the node where Osman is and the node that you sent. The i-th line is the response to your i-th query.

The testing system will allow you to read the response on the query only after your program print the query for the system and perform the `flush` operation.

Output

To make the queries your program must use standard output.

Your program must print the queries — A query consist of an integer number x_i ($1 \le x_i \le 2^n - 1$). Print one query per line (do not forget "*end of line*" after each x_i). After printing each line your program must perform operation `flush`.

In case your program find the node x where Osman is, print string "! x" (without quotes), where x is the answer, and terminate your program. An accepted solution should make at most 29 queries, printing the answer does not count as a query.

Example

standard input	standard output
2	
	1
1	
	3
2	
	2
0	
	! 2

Problem F. Free restricted flights

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

Alice and Bob are in the middle of a damn pandemic. They wanted to travel together somewhere for summer. However, they are in different countries and since the pandemic started, multiple flights have been canceled. What is worst, their respective countries do not allow tourists to enter, so they have to find another country to see each other. Nevertheless, they are allowed to use any country (including their residence countries) for stopovers.
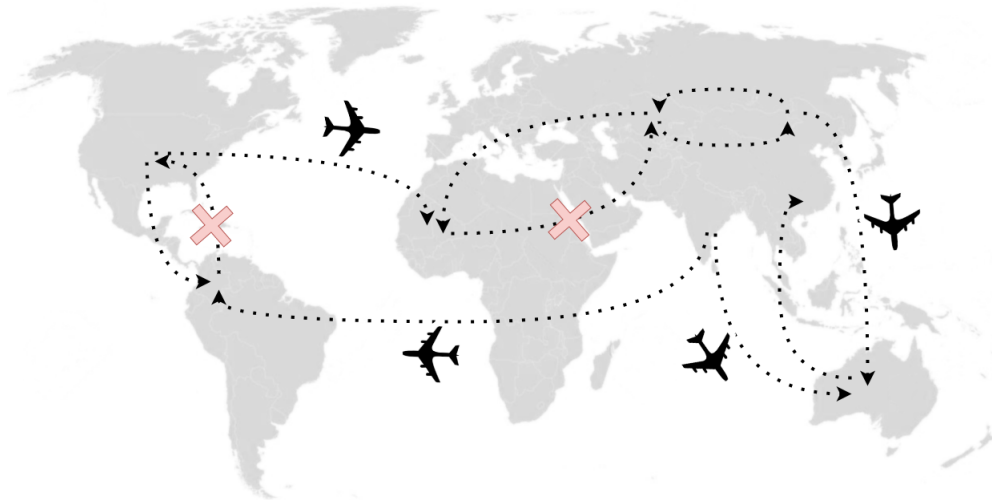
Alice and Bob have spent nearly all of their money on face masks, toilet paper, and disinfectant. Therefore, they need to minimize the total amount of money spent on flights, including the returning trip to their respective homes. Ironically, Alice and Bob are official members of the United Nihilistic Airlines League (UNAL), which has granted a special k-ticket to each of them. A k-ticket allows a passenger to take up to k free flights, those tickets are personal and non-transferable.

Can you help them to find the country where they can see each other with minimal cost?

Input

The first line consists of 2 integers separated by a space, n and m ($3 \leq n, m \leq 10^4$) — The number of countries, and the number of available flights, respectively.

The next line consists of 3 integers separated by a space, a, b and k ($0 \leq a, b < n$, $a \neq b$, $0 \leq k \leq 10$) — The country where Alice lives, the country where Bob lives, and the amount of free flights available on each k-ticket, respectively.

The next m lines contains 3 integers each: u, v ($0 \leq u, v < n$, $u \neq v$) and w ($1 \leq w \leq 10^3$) separated by a space, which represent a flight coming from u into v with cost w.

Output

Print two integers separated by a space, the index of the country where Alice and Bob should travel and the total round trip cost for both Alice and Bob. if there are many possible countries with minimal cost print the one with the lowest index.

If Alice and Bob cannot see each other print ">:(".

Examples

standard input	standard output
4 5 0 1 2 0 2 2 1 2 2 2 3 2 3 0 2 3 1 2	2 4
3 3 0 1 0 0 1 1 0 2 1 1 2 1	>:(
3 3 0 1 0 0 1 1 1 2 1 2 0 1	2 6

Problem G. Great dinner

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 megabytes

UNAL is about to restart classes and the leaders of the algorithms group want to receive all its members with a great dinner in the university campus central dining room.

The group has N members (excluding the leaders) numbered from 1 to N and the leaders want to organize them in a row before entering the dining room, in order to avoid any disorder. However, some students enjoy teasing others. In particular, there are M bullies and M victims, all of these $2 \times M$ students are different. Each bully annoys exactly one victim and each victim is being bullied by exactly one bully. To make dinner a great time for everyone, the leaders want to ensure that if student A annoys student B, B must not be ahead of A in the line.

Leaders enjoy math challenges, so they wonder how many ways can they organize all students on a line. Can you help them with that?

Input

The first line contains two integers N $(1 \le N \le 10^5)$ and M $(0 \le 2 \times M \le min(2 \times 10^3, N))$ separated by a space — The number of students, and the number of bullies and victims, respectively.

Each of the following M lines will contain two integers A and B $(1 \le A, B \le N)$ separated by a space, which means that student B must not be ahead of A in the line. It is guaranteed that each, A and B, appears at most once in the input.

Output

Output one integer, the number of ways the leaders can organize the students on a line modulo $10^9 + 7$.

Examples

standard input	standard output
4 2 2 1 4 3	6
4 1 1 3	12

Note

For the first case, there are only 6 ways to organize the students:

- 1 2 3 4

- 1 3 2 4

- 1 3 4 2

- 3 1 2 4

- 3 1 4 2

- 3 4 1 2

Problem H. Happy game

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

Teresa and Fabio are wandering in the UNAL campus waiting for Nick to get out of classes for lunch. They are getting bored, Teresa came up with a game.

They will play on the only street of the campus, the street is a straight line divided by labeled cells. The game consists of many rounds. In each round, both of them will select 2 different cells from the street. They will stand at each chosen cell watching to each other, then they will start walking to each other 1 step at a time until they meet at some point, either inside a cell or at the border of a cell.
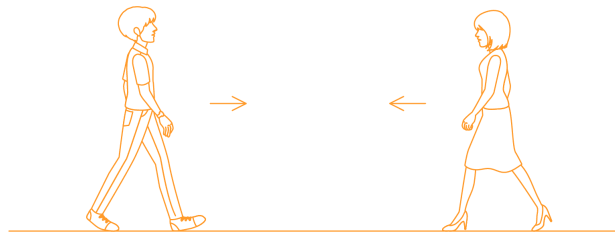
A round is called happy if at each step the label of both cells where they are standing is the same, and they meet inside a cell, not at the border.

Nick could go out of class at any moment, so to maximize their fun, they are not gonna do rounds that are not happy. Also, they don't want to repeat any round, two rounds are considered equal if the labels while walking are the same in both rounds for Teresa and Fabio. For example, if the labels on the street are *ababa* a round with Fabio selecting cell 1 and Teresa selecting cell 3 is equal to a round with Fabio selecting cell 5 and Teresa selecting cell 3.

Can you determine the maximum number of happy rounds they would do?

Input

The first line contains an integer n ($1 \leq n \leq 10^5$) — The number of cells on the street.

Following line will contains n lowercase english letters, which correspond to the labels of the street in order (from cell 1 to cell n).

Output

Print one integer that represents the maximum number of rounds the game could last.

Examples

standard input	standard output
1 c	0
3 aba	1
5 ababa	3
2 bb	0

Problem I. Incredible photography

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

The photography contest for the UNAL portfolio cover page is nearly over and Paula, as a big fan of photography, is worried since she still does not have a great picture to participate with. This is because this semester has been unusually harder than the previous one.

Paula is a fit girl. She usually goes to the GYM to be in shape but it seems that today she won't be able to go. That's why she decided that while she will be looking for the best picture, she will try to walk as much as she can.

Paula thinks that the best photo she can take will be at the top of some building. And she considers that it's not necessary to take a photo in a building whose height is less or equal than the height of the building where she is right now.

After Paula takes pictures at the top of a building, she chooses the next building to visit to take more photos. Naturally, she will choose only among the buildings that she is able to see. A building is in Paula's sight if and only if there is no other strictly higher building in between.

All Buildings in the UNAL are located in a single line, one after another. The distance that she is going to walk from one building to another is the absolute difference between the positions of the buildings in the line e.g. to go from building i to building j she needs to walk $|i - j|$ distance.

Paula is planning at which building she should start taking pictures. That's why she wants to know, for each building, what is the maximum distance she would walk if she starts to take pictures in that building. Can you help Paula with that?

Input

The first line contains an integer n ($1 \leq n \leq 10^5$) — The number of buildings.

The following line will contain n integers ($1 \leq h_i \leq 10^9$) — The height of the i-th building.

Output

Print n numbers on a line. The i-th number is the maximum distance Paula would walk if she starts taking pictures at building i.

Examples

standard input	standard output
4 3 1 2 4	3 6 5 0
1 15	0
5 3 3 1 5 5	4 3 6 0 0

Problem J. Java exam

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 megabytes

It's the end of the semester and students are really tired after having virtual classes. Next Monday guys of programming class have their final exam in Java. To avoid cheating, the teacher decided to allow them to take the exam in groups.

It is always difficult to form the groups because everybody has different preferences when choosing partners to work with. In particular, each student has a favorite partner. Of course, it is possible that a student doesn't like to work with anyone else, in this case, his favorite partner is himself.

Moreover, students have special partners, a student Z is a special partner of student X if $Z = X$ or if Z is a special partner of the favorite partner of X. Also this relation is anti-symmetric, i.e. if $Z \neq Y$ and Z is a special partner of X then X cannot be a special partner of Z.

Considering this, for 2 students X and Y to be in the same group the following conditions have to be met:

- X and Y are part of the group.

- If student Z is in the group then either the favorite partner of Z is in the group or Z is a special partner of all students in the group.

The Java exam will consist of a limited number of exercises, each one related to a unique topic taught during the semester. For each exercise, the teacher will choose randomly one member of the team and this student will explain the solution of the corresponding exercise. Of course, a student is able to explain a solution correctly if and only if the student knows perfectly the corresponding topic. The number of exercises explained correctly will be the grade of the exam for the whole group.

The teacher distinguishes very well his students and the topics they know, so he decided to get some statistics about the exam during today's class. Given students X and Y, he wants to know what would be the minimum grade that would get the smallest group containing those students. However, at the moment he decides to get a statistic, he only takes into account the students that are currently in class (yes, it's possible that some students arrive late or even go out early from class).

Since you got really bad scores on previous exams and you need extra credit, you decided to help your teacher to calculate the statistics during the whole class.

Input

The first line contains two integers n and b ($1 \leq n \leq 10^5$, $1 \leq b \leq 20$) — The number of students that are already in class and topics in the exam, respectively.

The following n blocks of input will describe the information for each student that is already in class. The i-th block consists of 2 lines described as follows:

The first line will contain two integers X_i and f_i meaning that student X_i is currently in class and f_i is his favorite partner. The second line starts with an integer ($1 \leq k_i \leq b$) which describes the number of topics student X_i knows. Next k_i integers will be the topics $t_k (1 \leq t_k \leq b)$ that student X_i knows perfectly.

Next line will contain integer q ($1 \leq q \leq 10^5$) — The number of events that occurs during the class.

The following q blocks of input will describe the information of each event happening during class. The i-th block starts with the type of event $type_i \in \{0, 1, 2\}$, then the rest of the input for that event goes as follows:

- if $type_i = 0$, it will be followed by one integer X_i meaning that student X_i left the class. It is guaranteed that this student is currently in class and once a student leaves, he never comes back.

- if $type_i = 1$, it will be followed by two integers X_i and Y_i, meaning that you want to calculate statistics for students X_i and Y_i. It is guaranteed that both students are currently in class.

- if $type_i = 2$, it will be followed by two integers X_i and f_i meaning that student X_i has just arrived to class and student f_i is his favorite partner. It is guaranteed that student X_i wasn't before in class. In this case, there is another line of input, it starts with an integer $(1 \le k_i \le b)$ which describes the number of topics student X_i knows. Next k_i integers will be the topics t_k $(1 \le t_k \le b)$ student X_i knows perfectly.

Each student is identified uniquely by an integer which does not exceed 10^9.

Note that it is possible for some students to never arrive to class.

Output

For each event of type 1 print the minimum grade that would get the smallest group with students X_i and Y_i. If a group can't be formed with the students in class print -1.

Examples

standard input	standard output
1 1 1 2 1 1 1 1 1 1	1
7 4 1 2 1 4 2 3 1 1 3 4 4 1 2 3 4 4 5 4 3 1 2 4 5 6 4 1 4 2 3 6 7 3 1 2 4 7 7 4 4 1 2 3 8 1 3 5 1 5 7 0 4 1 3 5 2 8 4 4 3 4 1 2 1 8 4 1 8 8 1 1 1	4 3 -1 -1 4 1

Problem K. Katastrophic sort

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 megabytes

Some students at the UNAL are learning how the computer works. So the great teacher is talking about how the files are sorted by name. Do you know how that works?. Well, they are ordered by the well known Katastrophic sort!.

Let's define Katastrophic sort: is an ordering of strings in lexicographical order, except that numbers are treated atomically, i.e., as if they were a single character.

Let's see an example If we sort strings **z11** and **z2** by lexicographical sorting we get:

$$z11 < z2$$

But if we sort them by Katastrophic sorting we get:

$$z2 < z11$$

Since both start by **"z"** the order is defined by the order of the numeric part. And as humans, we know that **2** is less than **11**.

In order to test your understanding, the teacher gives you two strings. You need to output which one is less than the other.

Input

You are given 2 lines. In each line one of the strings to compare. The length of each string is at most 10^5.

Each string will consist of a non-empty english lowercase alphabetic part followed with a non-empty numeric part

It is guaranteed that the numeric parts in the strings are integers and won't have leading zeroes.

It is guaranteed that the alphabetic part has the same length in both strings.

Output

Print $<$ if the first string is less than the second according to the Katastrophic sort. $>$ if the first string is greater than second according to the Katastrophic sort and $=$ otherwise.

Examples

standard input	standard output
z2 z11	<
abd14 abc14	>
asgfsd4213456 asgfsd4213456	=

Problem L. Lonely day

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 megabytes

Graduations at UNAL are coming and due to the quarantine and to avoid crowds, each student was scheduled to go individually to pick up their diploma on the UNAL campus.

Today it's Vitor's time, he is dressed in a fancy suit and ready to go. Let's imagine the university as a grid of N x M cells, where the lower-left corner is $(1, 1)$ and the upper-right corner is (N, M). Each cell has one of the following characters:

- "**.**" represents a clean cell. Vitor can step on it without ruining his suit

- "**X**" represents a dirty cell. Vitor can't step on it because it will ruin his suit.

- "**S**" represents the starting cell, where Vitor is at the beginning. It is also a clean cell. (There will be exactly one cell with this letter).

- "**E**" represents the final cell, where Vitor will receive his diploma. It is also a clean cell. (There will be exactly one cell with this letter).

Vitor can walk only through clean cells and for each step he can go from one to another cell if they share a side. However, UNAL keeps some secrets, one of them is the existence of tunnels that can take him from one cell to another cell in one step and keep him clean, isn't that magical? More precisely, a clean cell (x_1, y_1) is connected with another clean cell (x_2, y_2) by a tunnel if and only if the following conditions are met:

- $x_1 = x_2$ **OR** $y_1 = y_2$.

- When $x_1 = x_2$, each cell (x_1, y) must be a dirty cell, for all valid y in $min(y_1, y_2) < y < max(y_1, y_2)$.

- When $y_1 = y_2$, each cell (x, y_1) must be a dirty cell, for all valid x in $min(x_1, x_2) < x < max(x_1, x_2)$.

Vitor wants to reach his destination as soon as possible and remain clean at all times. He is asking you to help him accomplish this.

Input

The first line contains two integers N and M ($2 \leq N, M \leq 2000$) separated by a space.

The following N lines, each will contain a string with M characters.

Output

If there is no way to reach the destination print -1.

Otherwise, the output will consist of two lines. The first line will be an integer X where X is the minimum number of steps that will take Vitor to reach his destination. The second line will be a string with length X where the ith character describe the direction that Vitor should take in the ith step. The string only contains these characters:

- "L" describes a step from (x, y) to (a, y) where a < x

- "R" describes a step from (x, y) to (a, y) where a > x

- "D" describes a step from (x, y) to (x, b) where b < y

- "U" describes a step from (x, y) to (x, b) where b > y

Print the smallest lexicographical solution if there is more than one.

Examples

standard input	standard output
3 3 S.. ... ..E	4 DDRR
3 3 SX. XXX XXE	2 RD
2 2 SX XE	-1

Problem M. Magic spells

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 megabytes

The Ultra Nice Abracadabra List (UNAL) is a list of n magic spells for young student wizards in the school of magic, a magic spell is a sequence of English characters which can be used to make something magic, all magic spells have the incredible property of being subsequences of the ultimate magic spell s. Actually, all non-empty subsequences of s are spells, although not all are written in the UNAL.

Each student has its own copy of the UNAL when entering into the school, in particular, Andres, who is a new student, just got his copy of the UNAL. However, many of his mean (potential dark wizards) classmates took his copy of the UNAL and wrote some (possibly zero) characters after each of the spells in the list.

Andres is not dumb and he realized what his classmates did. He knows s and for him, it is enough for each line in the modified UNAL to get the longest prefix that is also a spell. Help Andres with this non-magic work.

Input

First line of input consists of the ultimate magic spell s, which is a string consisting of at most $2 * 10^5$ english lowercase characters.

Second line of input consists of a single integer n — The number of spells in the UNAL.

Next n ($1 \leq n \leq 10^5$) lines follow, the $i - th$ line consists of a non-empty string a_i — The $i - th$ spell of the UNAL with some (possibly zero) characters added at the end.

It is guaranteed that the total number of characters in the UNAL with modifications is at most $2 * 10^5$

Output

Output consists of n lines, each line should consists of the longest possible spell, if it is not possible to make a spell print "IMPOSSIBLE" (without quotes) instead.

Example

standard input	standard output
abracadabra	abra
5	cadabra
abra	abcd
cadabra	d
abcd	IMPOSSIBLE
dcba	
magic	

Problem N. Name this problem

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

Jolany is a little girl who lives in a faraway galaxy where the time travel (only to the future) is allowed. Today is Jolany's birthday and she received her first time travel machine as a gift from her parents.

In order to receive more gifts, she wants to use her new time machine to jump directly to her next birthday, exactly one year from now. A year on her planet has n days.

Time machines on her planet have two nice properties:

- A machine can only be used once per day

- A machine is non-deterministic and each day it will work with a different probability. If the machine works the time traveler will go directly to the desired date

Jolany set her machine to send her n days to the future and then she wrote down in a piece of paper a list with the probabilities that the machine works for each of the n days.

Unfortunately, she pushed the "hard mode" button in the machine, this button hides the array of probabilities to the user and then applies a random permutation to it, know the machine won't work with the original array but with a hidden permutation of it. Notice that there are always $n!$ different and equally probable permutations of the original array.

The only information Jolany has is the original array of probabilities and she wants to know the expected number of days she needs to wait until her next birthday. Can you help her?

Input

The first line contains integer n ($1 \le n \le 5000$) — The number of days in a year in Jolany's planet.

The second line contains n numbers — The original array of probabilities. Every probability is between 0 and 1 (inclusive) and has at most 4 digits after the decimal point.

Output

Output one line containing the answer to the task. Your answer will be considered correct if the absolute or relative error is at most 10^{-6}.

Examples

standard input	standard output
1 0.2	0.800000000000
2 0.5 1.0	0.250000000000
3 0.3 0.1 0.1	2.090333333333

Note

In the first example the machine will work with probability 0.2 otherwise Jonaly will need to wait one day so the answer is $0.2 * 0 + 0.8 * 1$.

In the second example there are 2 possible equally probable arrays [0.5, 1] with expected value 0.5 and [1, 0.5] with expected value 0. So, the answer is 0.25.