Problem A. Anisotropic Numbers

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

We will consider the positive integer N `anisotropic`, if its decimal notation, after rotation of **each** digit by 180 degrees, still represents some correct integer (the same or not).

For example, first four anisotropic numbers are 6 (9 after the flip), 8 (8 after the flip), 9 (6 after the flip) and 60 (90 after the flip).

The assembler for the new CPU "pdqb-69" is using only the anisotropic integers as the addresses.

Your task is to convert usual hexadecimal address into the anisotropic one, i.e. given the integer X, written in hexadecimal form, your task is to write X'th anisotropic integer.

Input

Input contains some positive integer X, written in the **hexadecimal** notation. The digits above 9 are represented by lowercase English letters '`a`'-'`f`' respectively. The string representing X does not contain the leading zeroes, $1 \leq N \leq 2^{65\,536}$.

Output

Print the X-th anisotropic integer in the **decimal** notation.

Examples

standard input	standard output
1	6
4	60
6	68

Problem B. Automatic Validation

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

You are given 10 integers. Each pair of integers is split either by single space, eithet by the single line feed.

You are preparing the contest for the Python users, so the type of the separating character is important.

Your task is to determine, it the integers are separated only with the spaces, only with the line feeds or each of two separators was used at least once.

Input

The input contains 10 positive integers a_i ($1 \le a_i \le 1000$). Each two adjacent integers are separated either by single space, either by single line feed. There is a single line feed after the last integer and there are no blank characters after the first integer.

Output

If the integers are separated by spaces, print 1. If the integers are separated by line feeds, print 2. If both spaces and line feeds are used, print 0.

Examples

standard input	standard output
10 83 600 6 3 9 1 9 9 7	1
8 1 27 64 125 216 343 512 729 1000	2

Problem C. Clusters

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

Ih the Byteland, there are N cities and $N-1$ bidirectional road connecting some pairs of the cities such as it is possible to move between cities using one of several roads. The cities are enumerated by sequential integers starting from 1.

The city 1 is the capital of Byteland. The region with the central city u is the subtree with root u in the rooted tree with the capital as root.

The mobile communications in city i are provided by the company c_i. Two cities are considered to be in one cluster, if it is possible to move between those cities in a way that all the cities on the path have the same mobile provider.

The Ministry of Information may issue Q orders in the form u c, it means that all cities in the region with u as the central city now have the same mobile provider c.

Your task is to print total number of clusters in Byteland after each order.

Input

The first line of the input contains two integers N and Q — number of cities in Byteland and number of the orders, respectively ($1 \le N, Q \le 2 \cdot 10^5$).

Each of the following $N-1$ lines contains two integers a and b ($1 \le a, b, \le N$) — the cities connected by some road. You may assume that it is posssible to move between any pair of the cities, using one of several roads.

The following line contains N integers c_i ($1 \le c_i \le N$) — IDs of the initial providers for the cities.

Each of the following M lines contains two integers u and c ($1 \le u, c \le N$) — the central city of the region where the new provider is forced and id of the provider.

Output

For each query print one integer — the total number of clusters after this query.

Example

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

Problem D. Dangerous Comet

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

At the some planet in the galaxy far far away, master Yoda several historical books has found.

There is a dangerous comet that is visible from this planet time by time. The scientists do not know the precise period X of that comet, but they are sure that it exists and the comet is visible the planet once per X years (i.e. if the comet was visible in year 2022, it will be visible again in year $2022 + X$ and previous time it was visible in year $2022 - X$ and so on).

Each of books found by Yoda contain two sequences of years: the years A_i when the comet was visible, and lot of disasters happen on the planet, and the years B_i when the comet was not visible.

For each book Yoda wants to check its consistency, i.e if it is possible to find an integer X such as the comet was visible in each of years A_i and was not visible in each of years B_i.

Input

The first line of the input contains one integer — the number of the books T ($1 \le T \le 5$).

The description of each book starts with two integers n and m – the lengths of the sequences A and B, respectively ($1 \le n, m \le 7.5 \cdot 10^4$). The second line of the description contains n integers A_i ($1 \le A_i \le 10^{18}$) — the years when the comet was visible, according with that book. The third line of the description contains m integers B_i ($1 \le B_i \le 10^{18}$) — the years when the comet was invisible, according with that book.

Output

For each book print at the new line '`1`', if it is consistent, and '`0`' otherwise.

Example

standard input	standard output
4 4 3 1 7 4 13 3 11 9 3 2 9 5 3 7 11 2 3 3 15 1 7 2 2 2 5643634654354 12346544323565 22341124534 7655867344	1 0 1 1

Problem E. Easy Transformations

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

Bytica found a matrix of n rows and n columns. The rows are numbered starting from zero from top to bottom, and the columns are numbered starting from zero from left to right. The cell in the intersection of the i-th row and the j-th column is denoted as (i, j). For each cell (i, j), there is an integer $i \times n + j$ written in.

Bytica is going to perform q successive transformations. The transformations are of two possible types.

The i-th transformation is of t_i-th type, and it's described by 3 parameters l_i, r_i, d_i.

If $t_i = 1$, the number in cell $(x, (y + d_i) \bmod n)$ where $l_i \le x \le r_i, 0 \le y < n$ will be transferred to the cell (x, y) by the transformation.

If $t_i = 2$, the number in cell $((x + d_i) \bmod n, y)$ where $0 \le x < n, l_i \le y \le r_i$ will be transferred to the cell (x, y) by the transformation.

Note that $a \bmod b$ means the remainder of a after division by b.

Bytica would like to know the final configuration of the matrix.

Input

The first line of the input contains two integers n, q ($1 \le n \le 200$, $1 \le q \le 10^5$).

The i-th of the following q lines contains 4 integers t_i, l_i, r_i, d_i ($t_i \in \{1, 2\}, 0 \le l_i \le r_i < n, 0 \le d_i < n$).

Output

Print n lines. The i-th of those lines shall contain n integers $a_{i,0}, a_{i,1}, \ldots, a_{i,n-1}$ denoting the final number in cell (i, j) respectively.

Examples

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

Problem F. Four Players Game

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

Byteazar is playing the game for 4 players. Initially he have N Byteland talers. Three other players have an infinite bank.

Each round of the game each player bets 1 Byteland taler. The round ends up with the victory of one player, who gets all 4 Byteland talers for this round.

Calculate the probability that Byteasar will lose all his money if he will play this game infinitely and if his probability to win one round is p.

Input

First line of the input contains one integer N ($1 \le N \le 100$) — the initial amount of money for Byteasar. Second line contains one real number p ($0.25 < p < 1$) — the winning probability for Byteasar (he is skilled player, so his chance to win is greater than just 0.25).

Output

Print the answer with absolute or relative error 10^{-4} or better.

Examples

standard input	standard output
10 0.400000	0.02496787396946
1 0.836006	0.16460777726365

Problem G. Mystical Number

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

Scientists are discovered that the monetary system of ancient Byteland had the following property:

- All coins has **pairwise distinct** integer values.
- Each amount of money from 1 to the mystical number B inclusively can be represented by a set of pairwise different coins, and **in a unique way**

In neighboring ancient Berland, the similar coins were used, but this rule was not held.

Archaeologists have found the box with N ancient Berland or ancient Byteland coins. It is known that at least one of each Ancient Byteland coins can be found among these coins. Your task is to find the maximum possible value of the mystical number B.

Input

The first line of the input contains one integer N ($1 \le N \le 10^5$) — the number of the coins found. The second line contains N integers between 1 and 10^{18}, inclusively — the values of the coins.

Output

Print one integer — the maximally possible value for the mystical integer B. You may assume that for the input data such $B > 0$ exists.

Examples

standard input	standard output
3 5 1 3	1
3 2 3 1	3

Problem H. Hidden Permutation

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

This is an interactive problem

There is the a hidden permutation P of length 2^N (i.e. sequence of length 2^N, containing each integer from 1 to 2^N exactly once).

You may use the following query:

The query is sent as the integer sequence Q of length 2^N, consisting only from integers between 1 and 2^N (but those integers can be repeating).

The jury program will build the answer sequence A of length 2^N in the following way:

- Initially all A_i are equal to 0.
- For each i ($1 \le i \le 2^N$) when $P_{Q_i} > P_i$, A_{Q_i} is increased by 1.

Then your program receives the generated answer sequence.

Your task is correctly guess the permutation P, using no more than 60 queries.

Interaction Protocol

At the beginning of the interaction, the jury program tells to you an integer N ($1 \le N \le 10$).

The query have the following format:

`?` Q_1 Q_2 ... Q_{2^N}

where $1 \le Q_i \le 2^N$.

The jury program will then print the answer sequence as 2^N space-separated integers A_i, in order.

When you are ready to tell the permutation P, print

`!` P_1 P_2 ... P_{2^N}.

This is not counted as query.

Example

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

Note

Do not forget to print end-of-line after last integer in each query or in the final answer, and flush the output buffer after each query/final answer. Otherwise your solution may have the WTL error.

Problem I. Hard Game

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

This is an interactive problem

At the beginning of the game, $2n$ consecutive integers are written on board. Alice and Bob make their moves in a turn with Alice starting first. The move consists of erasing one of integers that still are on board. After $2n-2$ turns, there will remain only two integers. If their GCD is not equal to 1, then Alice wins, otherwise Bob wins.

Bob wants to beat Alice in this game, and asks you to help him with the program that can play for him.

Interaction Protocol

At the beginning, the jury program tells you one integer n ($1 \le n \le 10^5$), that defines the size of array.

Initially, at the board are $2n$ integers.

Then $n-1$ times the following two actions happen: the jury program prints one integer between 1 and $2n$ — the integer that is erased by Alice, and your program shall answer with some non-erased integer that is erased by Bob. Attempt to erase already erased integer immediately causes Wrong Answer error.

If the greater common divisor of the remaining two integers is 1, you win. Otherwise you lose and recieve Wrong Answer.

Examples

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

Note

Do not forget to end each your turn with end-of-line character and to flush the buffer after it.

Problem J. Joking With Wizard

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

This is an interactive problem.

...In tonight's nightmare you talked with a wizard... one bad joke, and found self at the chessboard as some white piece. You can see that you are placed on e2, but you cannot see, who you are — the Pawn, the Rook, the Knight, the Bishop, the Queen or the King. You can't ask the other pieces, because you are alone on the board.

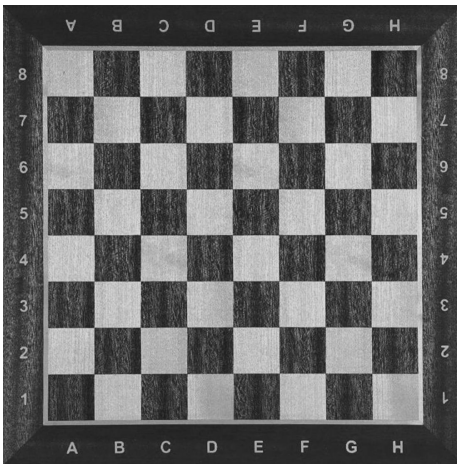
You decided to choose a cell and try to move to this cell. If this move is legal for your piece, you are moving to this cell, otherwise you stay where you are. You are planning to do no more than 4 attempts to determine your piece and then **return to the same field e2** — else the angry wizard can turn you into something worse than chess piece...

Note that the white chess pieces are moving in the following way:

- Pawn — one cell forward (i.e. keep the column and increment the row by 1.
- Rook — at arbitrary number of cells at same row or same column.
- Bishop — diagonally on all four diagonal directions.
- Knight — moving two cells at one direction and one cell in the any of two orthogonal directions, in L-shape.
- Queen — at arbitrary number of cells in same row, column or diagonal directions.
- King — at any of 8 cells that share a common point with the current.

More complicated chess rules related to the pawns (like double first move or transformation to other piece after reaching cell at topmost row) are `not applied` here.

The cells are denoted by the 2-character strings: the first character — the letter between '`a`' and '`h`', inclusively — denote the column, the second character — the digit between '`1`' and '`8`', inclusively — denote the row.



Interaction Protocol

The interaction is started by your program, printing the attempt in the format `?` *target*, where the *target* is the cell in the chess notation. If you succeed, you receive the integer 1 (and now are on cell *target*), if you failed, you receive 0 and did not move anywhere.

If you want to print the answer, print '`!`' c, where c is the lowercase English character denoting the piece: '`p`' for the Pawn, '`r`' for the Rook, '`b`' for the Bishop, '`n`' for the Knight, '`q`' for the Queen and '`k`' for the King. You may do that exactly once; ensure that you are on the cell e2 when you are printing the answer. This action **does not** counted as the query.

Example

standard input	standard output
0 1 0 1 1	? g3 ? e3 ? d2 ? e5 ? e2 ! r

Note

The interaction in the sample is only for illustration of the process: 5 queries are too much for the correct solution.

Problem K. Knights of Light and Darkness

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

This is an interactive problem

There are N knights on the island. Each knight at any moment of time is either Knight of Light or Knight of Darkness. Knights of Light answer truth on any question, while Knights of Darkness lie on any question, i.e. answer "Yes" instead "No" and "No" instead of "Yes".

When **any** knight answers "Yes", he **changes** his align immediately after the answer, i.e. the Knight of Light become the Knight of Darkness and the Knight of Darkness become the Knight of Light.

You are sent to the island with the important secret mission: tell the number of Knights of Light at the moment of your **departure** from the island.

To get the information, you may ask any person about any **other** person (the knights are enumerated by the sequential integers between 1 and N) in the form "Is knight Y the Knight of Light?" or "Is knight Y the Knight of Darkness?". You cannot ask the knight about himself, because it will look too suspiciously.

Can you finish this task for the finite number of questions? If yes, ask the **minimal possible** number of questions and then tell current number of the Knights of Light.

Note that the jury have the proof that for any position where the solution exists defined some optimal number of questions to solve this task,

Interaction Protocol

At the beginning of the interaction, you receive one integer N ($1 \le N \le 1000$) — the number of the knights on the island.

Then you may ask the questions.

If you want to ask the knight X, is the knight Y the Knight of Light, use the query in form "`? L` X Y".

If you want to ask the knight X, is the knight Y the Knight of Darkness, use the query in form "`? D` X Y". X and Y are the integers between 1 and N.

If you after several questions (or immediately) decided that it was impossible to perform your mission, print the message "`!` -1" and exit.

If you in some moment decided that you know the current number of the Knights of Light, print the message "`!` N_l", where N_l is the current number of Knights of Light, and exit.

Note that interactor is **adaptive**, i.e. it may generate the initial distribution accordingly with your questions.

In case when you decide that the mission is impossible, you can ask no more than $4N/3$ questions before you do that. If you are going to tell the answer, you shall ask the **mininal possible** number of the questions.

Example

standard input	standard output
3 0 1 0	 ? L 1 2 ? D 1 2 ? D 3 1 ! 0

Note

Do not forget to print end-of-line after last integer in each query or in the final answer, and flush the output buffer after each query/final answer. Otherwise your solution may have the WTL error.

Note that the sample interaction in the statement is **only for the illustration of format** — the person who asked questions may have no reasons to answer and he is succeed (if he is) only on the blind luck.

Problem L. Decimally Full Squares

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

Consider **positive** integer x *decimally full square*, if it is the square of some integer and if any suffix of its decimal notation is the square of some integer.

For example, consider 1064, its suffixes are 4, 64, 064, they all are the squares (of 2, 8 and 8 respectively; when the suffix starts from the leading zero, that zero is ignored), but 1064 itself is not a square, so 1064 is not decimally full square.

Given an integer N, your task is to count number of the decimally full squares does not exceeding N.

Input

The first line of the input contains one integer N ($1 \le N \le 10^{100}$).

Output

Print one integer — the number of decimally full squares, does not exceeding N.

Examples

standard input	standard output
5	2
70	5

Problem M. Monotonous Palindrome

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

Given the string S, consisting of lowercase English letters.

The substring of the string S is the contiguous part of S, starting from some position i and ending on some position j. If i and j coincide witn the first and last position, the substring is the string itself, if $i = j$, then substring consists of one character. For example, the string "`pc`" is the substring of the string "`icpc`", ans substring "`cc`" is not.

Your task is to find the longest substring of the given string such as two following conditions are held **simultaneously**:

- The string is a palindrome (i.e is equal to itself being read backwards).
- The string is monotonous, i.e. for any character in the string the previous character (if it exists) is not greater than current one.

If there are more than one substrings of the maximal lenght that conform those condtions, choose lexicographically maximal one (i.e. one that will be last in the dictionary).

Input

Input contains one string S, $1 \le |S| \le 10^5$, the string consists of the lowercase English letters.

Output

Print one string — the answer to the problem.

Examples

standard input	standard output
xxx	xxx
abcd	d