Codeforces Round #705 (Div. 2)

A. Anti-knapsack

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

You are given two integers n and k. You are asked to choose maximum number of distinct integers from 1 to n so that there is no subset of chosen numbers with sum equal to k.

A subset of a set is a set that can be obtained from initial one by removing some (possibly all or none) elements of it.

Input
The first line contains the number of test cases T ($1 \le T \le 100$).

Each of the next T lines contains two integers n and k ($1 \le k \le n \le 1000$) — the description of test cases.

Output
For each test case output two lines. In the first line output a single integer m — the number of chosen integers.

In the second line output m distinct integers from 1 to n — the chosen numbers.

If there are multiple answers, print any. You can print the numbers in any order.

Example

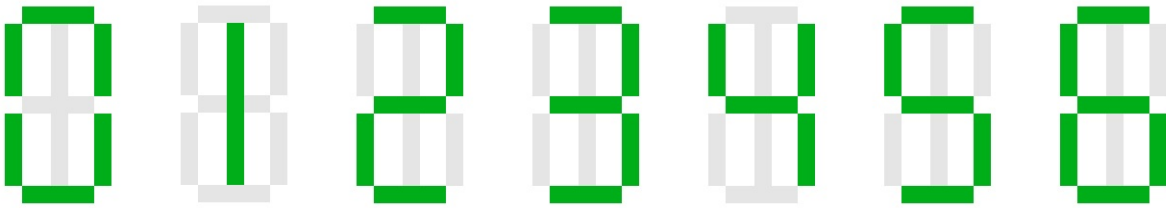
input
3 3 2 5 3 1 1

output
2 3 1 3 4 5 2 0

B. Planet Lapituletti

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

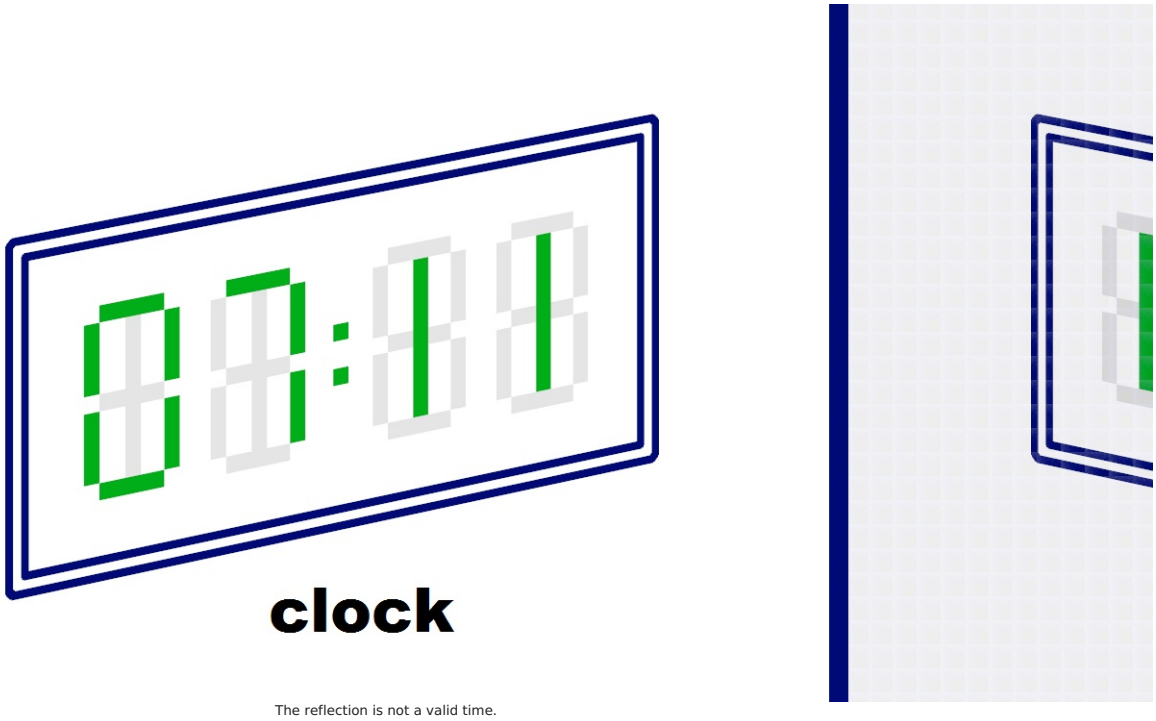
The time on the planet Lapituletti goes the same way it goes on Earth but a day lasts h hours and each hour lasts m minutes. The inhabitants of that planet use digital clocks similar to earth ones. Clocks display time in a format `HH:MM` (the number of hours in decimal is displayed first, then (after the colon) follows the number of minutes in decimal; the number of minutes and hours is written with leading zeros if needed to form a two-digit number). Hours are numbered from 0 to $h-1$ and minutes are numbered from 0 to $m-1$.



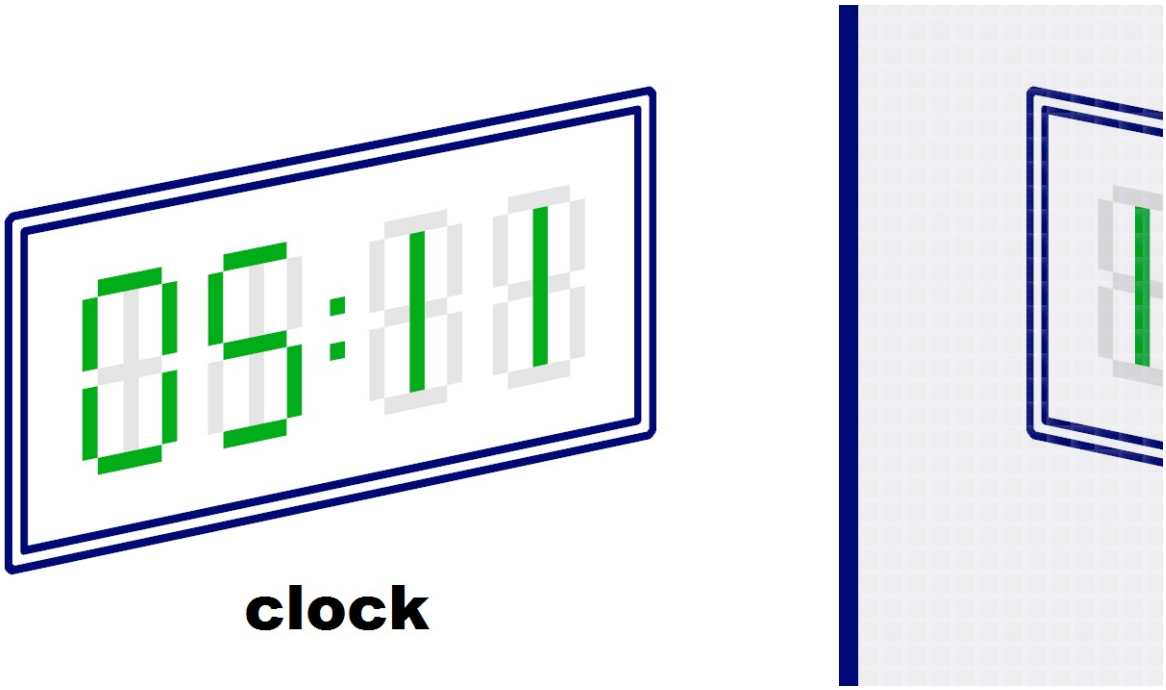
That's how the digits are displayed on the clock. Please note that digit 1 is placed in the **middle** of its position.

A standard mirror is in use on the planet Lapituletti. Inhabitants often look at the reflection of the digital clocks in the mirror and feel happy when what you see on the reflected clocks is a valid time (that means that you see valid digits in the reflection and this time can be seen on the normal clocks at some moment of a day).

The image of the clocks in the mirror is reflected against a vertical axis.



The reflection is not a valid time.



The reflection is a valid time with $h = 24$, $m = 60$. However, for example, if $h = 10$, $m = 60$, then the reflection is not a valid time.

An inhabitant of the planet Lapituletti begins to look at a mirrored image of the clocks at some time moment s and wants to know the nearest future time moment (which can possibly happen on the next day), when the reflected clock time is valid.

It can be shown that with any h, m, s such a moment exists. If the reflected time is correct at the moment the inhabitant began to look at the clock, that moment is considered the nearest.

You are asked to solve the problem for several test cases.

Input
The first line contains a single integer T ($1 \le T \le 100$) — the number of test cases.

The next $2 \cdot T$ lines contain the description of test cases. The description of each test case consists of two lines.

The first line of a test case contains two integers h, m ($1 \le h, m \le 100$).

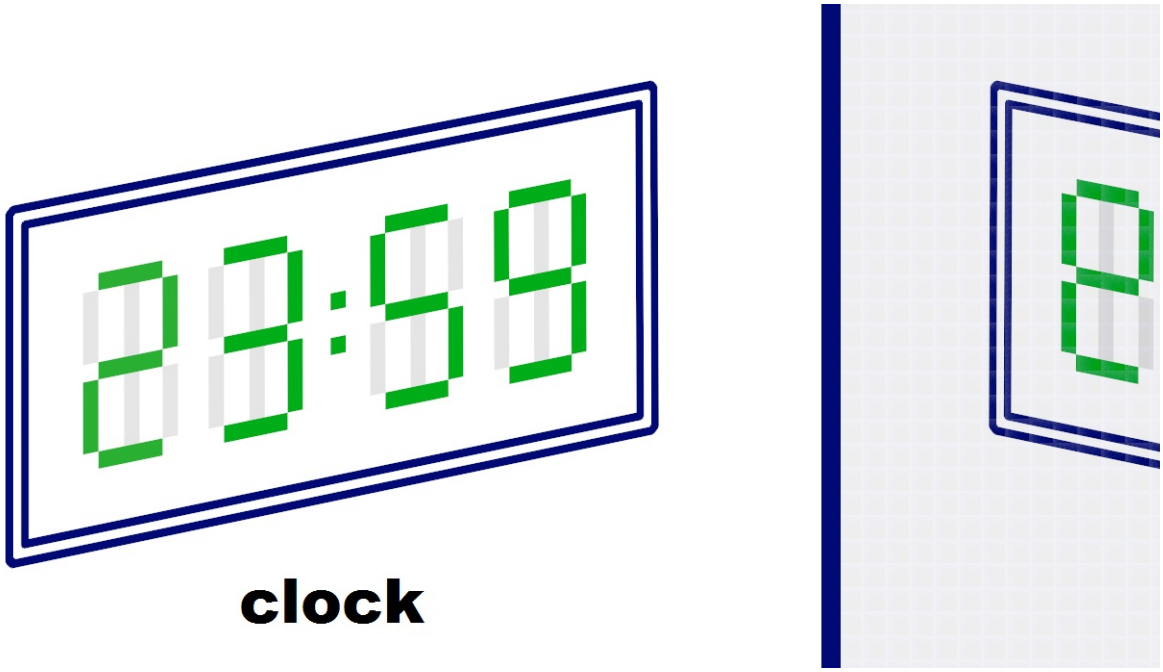
The second line contains the start time s in the described format `HH:MM`.

Output
For each test case output in a separate line the nearest moment in format `HH:MM` when the reflected time is correct.

Example

input
5 24 60 12:21 24 60 23:59 90 80 52:26 1 100 00:01 10 10 04:04
output
12:21 00:00 52:28 00:00 00:00

Note
In the second test case it is not hard to show that the reflection of `23:59` is incorrect, while the reflection of the moment `00:00` on the next day is correct.



C. K-beautiful Strings

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

You are given a string s consisting of lowercase English letters and a number k. Let's call a string consisting of lowercase English letters *beautiful* if the number of occurrences of each letter in that string is divisible by k. You are asked to find the lexicographically smallest beautiful string of length n, which is lexicographically greater or equal to string s. If such a string does not exist, output -1.

A string a is lexicographically smaller than a string b if and only if one of the following holds:

- a is a prefix of b, but $a \neq b$;
- in the first position where a and b differ, the string a has a letter that appears earlier in the alphabet than the corresponding letter in b.

Input

The first line contains a single integer T ($1 \leq T \leq 10\,000$) — the number of test cases.

The next $2 \cdot T$ lines contain the description of test cases. The description of each test case consists of two lines.

The first line of the description contains two integers n and k ($1 \leq k \leq n \leq 10^5$) — the length of string s and number k respectively.

The second line contains string s consisting of lowercase English letters.

It is guaranteed that the sum of n over all test cases does not exceed 10^5.

Output

For each test case output in a separate line lexicographically smallest beautiful string of length n, which is greater or equal to string s, or -1 if such a string does not exist.

Example

input
4 4 2 abcd 3 1 abc 4 3 aaaa 9 3 abaabaaaa

output
acac abc -1 abaabaaab

Note

In the first test case "acac" is greater than or equal to s, and each letter appears 2 or 0 times in it, so it is beautiful.

In the second test case each letter appears 0 or 1 times in s, so s itself is the answer.

We can show that there is no suitable string in the third test case.

In the fourth test case each letter appears 0, 3, or 6 times in "abaabaaab". All these integers are divisible by 3.

D. GCD of an Array

time limit per test: 2.5 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

You are given an array a of length n. You are asked to process q queries of the following format: given integers i and x, multiply a_i by x.

After processing each query you need to output the greatest common divisor (GCD) of all elements of the array a.

Since the answer can be too large, you are asked to output it modulo $10^9 + 7$.

Input

The first line contains two integers — n and q ($1 \leq n, q \leq 2 \cdot 10^5$).

The second line contains n integers $a_1, a_2, \ldots, a_n$ ($1 \leq a_i \leq 2 \cdot 10^5$) — the elements of the array a before the changes.

The next q lines contain queries in the following format: each line contains two integers i and x ($1 \leq i \leq n$, $1 \leq x \leq 2 \cdot 10^5$).

Output

Print q lines: after processing each query output the GCD of all elements modulo $10^9 + 7$ on a separate line.

Example

input

```
4 3
1 6 8 12
1 12
2 3
3 3
```

output

```
2
2
6
```

Note

After the first query the array is $[12, 6, 8, 12]$, $\gcd(12, 6, 8, 12) = 2$.

After the second query — $[12, 18, 8, 12]$, $\gcd(12, 18, 8, 12) = 2$.

After the third query — $[12, 18, 24, 12]$, $\gcd(12, 18, 24, 12) = 6$.

Here the $\gcd$ function denotes the greatest common divisor.

E. Enormous XOR

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

You are given two integers l and r in binary representation. Let $g(x, y)$ be equal to the bitwise XOR of all integers from x to y inclusive (that is $x \oplus (x+1) \oplus \cdots \oplus (y-1) \oplus y$). Let's define $f(l, r)$ as the maximum of all values of $g(x, y)$ satisfying $l \le x \le y \le r$.

Output $f(l, r)$.

Input

The first line contains a single integer n ($1 \le n \le 10^6$) — the length of the binary representation of r.

The second line contains the binary representation of l — a string of length n consisting of digits 0 and 1 ($0 \le l < 2^n$).

The third line contains the binary representation of r — a string of length n consisting of digits 0 and 1 ($0 \le r < 2^n$).

It is guaranteed that $l \le r$. The binary representation of r does not contain any extra leading zeros (if $r = 0$, the binary representation of it consists of a single zero). The binary representation of l is preceded with leading zeros so that its length is equal to n.

Output

In a single line output the value of $f(l, r)$ for the given pair of l and r in binary representation without extra leading zeros.

Examples

input

```
7
0010011
1111010
```

output

```
1111111
```

input

```
4
1010
1101
```

output

```
1101
```

Note

In sample test case $l = 19$, $r = 122$. $f(x, y)$ is maximal and is equal to 127, with $x = 27$, $y = 100$, for example.

F. Enchanted Matrix

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

This is an interactive problem.

There exists a matrix a of size $n \times m$ (n rows and m columns), you know only numbers n and m. The rows of the matrix are numbered from 1 to n from top to bottom, and columns of the matrix are numbered from 1 to m from left to right. The cell on the intersection of the x-th row and the y-th column is denoted as (x, y).

You are asked to find the number of pairs (r, c) ($1 \le r \le n$, $1 \le c \le m$, r is a divisor of n, c is a divisor of m) such that if we split the matrix into rectangles of size $r \times c$ (of height r rows and of width c columns, each cell belongs to exactly one rectangle), all those rectangles are pairwise equal.

You can use queries of the following type:

- ? h w i_1 j_1 i_2 j_2 ($1 \le h \le n$, $1 \le w \le m$, $1 \le i_1, i_2 \le n$, $1 \le j_1, j_2 \le m$) — to check if **non-overlapping** subrectangles of height h rows and of width w columns of matrix a are equal or not. The upper left corner of the first rectangle is (i_1, j_1). The upper left corner of the second rectangle is (i_2, j_2). Subrectangles overlap, if they have at least one mutual cell. If the subrectangles in your query have incorrect coordinates (for example, they go beyond the boundaries of the matrix) or overlap, your solution will be considered incorrect.

You can use at most $3 \cdot \lfloor \log_2 (n + m) \rfloor$ queries. All elements of the matrix a are fixed before the start of your program and do not depend on your queries.

Input

The first line contains two integers n and m ($1 \le n, m \le 1000$) — the number of rows and columns, respectively.

Output

When ready, print a line with an exclamation mark ('`!`') and then the answer — the number of suitable pairs (r, c). After that your program should terminate.

Interaction

To make a query, print a line with the format "? h w i_1 j_1 i_2 j_2 ", where the integers are the height and width and the coordinates of upper left corners of non-overlapping rectangles, about which you want to know if they are equal or not.

After each query read a single integer t (t is 0 or 1): if the subrectangles are equal, $t = 1$, otherwise $t = 0$.

In case your query is of incorrect format or you asked more than $3 \cdot \lfloor \log_2 (n + m) \rfloor$ queries, you will receive the `Wrong Answer` verdict.

After printing a query do not forget to output end of line and flush the output. Otherwise, you will get `Idleness limit exceeded`. To do this, use:

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

It is guaranteed that the matrix a is fixed and won't change during the interaction process.

Hacks format

For hacks use the following format.

The first line contains two integers n and m ($1 \leq n, m \leq 1000$) — the number of rows and columns in the matrix, respectively.

Each of the next n lines contains m integers — the elements of matrix a. All the elements of the matrix must be integers between 1 and $n \cdot m$, inclusive.

Example

input
3 4 1 1 1 0

output
? 1 2 1 1 1 3 ? 1 2 2 1 2 3 ? 1 2 3 1 3 3 ? 1 1 1 1 1 2 ! 2

Note

In the example test the matrix a of size 3×4 is equal to:

1 2 1 2
3 3 3 3
2 1 2 1