Codeforces Round #449 (Div. 2)

A. Scarborough Fair

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

Are you going to Scarborough Fair?
Parsley, sage, rosemary and thyme.

Remember me to one who lives there.

He once was the true love of mine.

Willem is taking the girl to the highest building in island No.28, however, neither of them knows how to get there.

Willem asks his friend, Grick for directions, Grick helped them, and gave them a task.

Although the girl wants to help, Willem insists on doing it by himself.

Grick gave Willem a string of length n.

Willem needs to do m operations, each operation has four parameters l, r, c_1, c_2, which means that all symbols c_1 in range $[l, r]$ (from l-th to r-th, including l and r) are changed into c_2. String is 1-indexed.

Grick wants to know the final string after all the m operations.

Input

The first line contains two integers n and m $(1 \leq n, m \leq 100)$.

The second line contains a string s of length n, consisting of lowercase English letters.

Each of the next m lines contains four parameters l, r, c_1, c_2 $(1 \leq l \leq r \leq n$, c_1, c_2 are lowercase English letters), separated by space.

Output

Output string s after performing m operations described above.

Examples

input
3 1 ioi 1 1 i n
output
noi

input
5 3 wxhak 3 3 h x 1 5 x a 1 3 w g
output
gaaak

Note

For the second example:

After the first operation, the string is `wxxak`.

After the second operation, the string is `waaak`.

After the third operation, the string is `gaaak`.

B. Chtholly's request

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

— Thanks a lot for today.
— I experienced so many great things.

— You gave me memories like dreams... But I have to leave now...

— One last request, can you...

— Help me solve a Codeforces problem?

— ......

— What?

Chtholly has been thinking about a problem for days:

If a number is *palindrome* and length of its decimal representation without leading zeros is even, we call it a zcy number. A number is *palindrome* means when written in decimal representation, it contains no leading zeros and reads the same forwards and backwards. For example `12321` and `1221` are palindromes and `123` and `12451` are not. Moreover, `1221` is zcy number and `12321` is not.

Given integers k and p, calculate the sum of the k smallest zcy numbers and output this sum modulo p.

Unfortunately, Willem isn't good at solving this kind of problems, so he asks you for help!

Input
The first line contains two integers k and p ($1 \le k \le 10^5$, $1 \le p \le 10^9$).

Output
Output single integer — answer to the problem.

Examples

input
2 100
output
33

input
5 30
output
15

Note
In the first example, the smallest zcy number is 11, and the second smallest zcy number is 22.

In the second example, $(11 + 22 + 33 + 44 + 55) \bmod 30 = 15$.

C. Nephren gives a riddle

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

What are you doing at the end of the world? Are you busy? Will you save us?

Nephren is playing a game with little leprechauns.

She gives them an infinite array of strings, $f_{0...\infty}$.

f_0 is "`What are you doing at the end of the world? Are you busy? Will you save us?`".

She wants to let more people know about it, so she defines $f_i =$ "`What are you doing while sending "`f_{i-1}"`? Are you busy? Will you send "`f_{i-1}"`?`" for all $i \geq 1$.

For example, f_1 is

"`What are you doing while sending "What are you doing at the end of the world? Are you busy? Will you save us?"? Are you busy? Will you send "What are you doing at the end of the world? Are you busy? Will you save us?"?`". Note that the quotes in the very beginning and in the very end are for clarity and are not a part of f_1.

It can be seen that the characters in f_i are letters, question marks, (possibly) quotation marks and spaces.

Nephren will ask the little leprechauns q times. Each time she will let them find the k-th character of f_n. The characters are indexed starting from 1. If f_n consists of less than k characters, output '`.`' (without quotes).

Can you answer her queries?

Input

The first line contains one integer q ($1 \leq q \leq 10$) — the number of Nephren's questions.

Each of the next q lines describes Nephren's question and contains two integers n and k ($0 \leq n \leq 10^5$, $1 \leq k \leq 10^{18}$).

Output

One line containing q characters. The i-th character in it should be the answer for the i-th query.

Examples

input
3 1 1 1 2 1 111111111111
output
Wh.

input
5 0 69 1 194 1 139 0 47 1 66
output
abdef

input
10 4 1825

```
3 75
3 530
4 1829
4 1651
3 187
4 584
4 255
4 774
2 474
```

output

```
Areyoubusy
```

Note

For the first two examples, refer to f_0 and f_1 given in the legend.

D. Ithea Plays With Chtholly

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

This is an interactive problem. Refer to the Interaction section below for better understanding.

Ithea and Chtholly want to play a game in order to determine who can use the kitchen tonight.

Initially, Ithea puts n clear sheets of paper in a line. They are numbered from 1 to n from left to right.

This game will go on for m rounds. In each round, Ithea will give Chtholly an integer between 1 and c, and Chtholly needs to choose one of the sheets to write down this number (if there is already a number before, she will erase the original one and replace it with the new one).

Chtholly wins if, at any time, all the sheets are filled with a number and the n numbers are in non-decreasing order looking from left to right from sheet 1 to sheet n, and if after m rounds she still doesn't win, she loses the game.

Chtholly really wants to win the game as she wants to cook something for Willem. But she doesn't know how to win the game. So Chtholly finds you, and your task is to write a program to receive numbers that Ithea gives Chtholly and help her make the decision on which sheet of paper write this number.

Input

The first line contains 3 integers n, m and c ($n, m \geq 2, 1 \leq c \leq 1000, 1 \leq n \cdot \lceil \frac{c}{2} \rceil \leq m \leq 1000, \lceil x \rceil$ means x rounded up) — the number of sheets, the number of rounds and the largest possible number Ithea can give to Chtholly respectively. The remaining parts of input are given throughout the interaction process.

Interaction

In each round, your program needs to read one line containing a single integer p_i ($1 \leq p_i \leq c$), indicating the number given to Chtholly.

Your program should then output a line containing an integer between 1 and n, indicating the number of sheet to write down this number in.

After outputting each line, don't forget to flush the output. For example:

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

If Chtholly wins at the end of a round, no more input will become available and your program should terminate normally. It can be shown that under the constraints, it's always possible for Chtholly to win the game.

Example

input
2 4 4 2 1 3
output
1 2 2

Note

In the example, Chtholly initially knew there were 2 sheets, 4 rounds and each number was between 1 and 4. She then received a 2 and decided to write it in the 1st sheet. Then she received a 1 and wrote it in the 2nd sheet. At last, she received a 3 and replaced 1 with 3 in the 2nd sheet. At this time all the sheets were filled with a number and they were non-decreasing, so she won the game.

Note that it is required that your program terminate immediately after Chtholly wins and do not read numbers from the input for the remaining rounds. If not, undefined behaviour may arise and it won't be sure whether your program will be accepted or rejected. Also

because of this, please be careful when hacking others' codes. In the sample, Chtholly won the game after the 3rd round, so it is required that your program doesn't read the number of the remaining 4th round.

The input format for hacking:

- The first line contains 3 integers n, m and c;
- The following m lines each contains an integer between 1 and c, indicating the number given to Chtholly in each round.

E. Willem, Chtholly and Seniorious

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

— Willem...

— What's the matter?

— It seems that there's something wrong with Seniorious...

— I'll have a look...

Seniorious is made by linking special talismans in particular order.

After over 500 years, the carillon is now in bad condition, so Willem decides to examine it thoroughly.

Seniorious has n pieces of talisman. Willem puts them in a line, the i-th of which is an integer a_i.

In order to maintain it, Willem needs to perform m operations.

There are four types of operations:

- $1\ l\ r\ x$: For each i such that $l \le i \le r$, assign $a_i + x$ to a_i.
- $2\ l\ r\ x$: For each i such that $l \le i \le r$, assign x to a_i.
- $3\ l\ r\ x$: Print the x-th smallest number in the index range $[l, r]$, i.e. the element at the x-th position if all the elements a_i such that $l \le i \le r$ are taken and sorted into an array of non-decreasing integers. It's guaranteed that $1 \le x \le r - l + 1$.
- $4\ l\ r\ x\ y$: Print the sum of the x-th power of a_i such that $l \le i \le r$, modulo y, i.e. $(\sum_{i=l}^{r} a_i^x) \bmod y$.

Input

The only line contains four integers n, m, $seed$, v_{max} ($1 \le n, m \le 10^5$, $0 \le seed < 10^9 + 7$, $1 \le vmax \le 10^9$).

The initial values and operations are generated using following pseudo code:

```
def rnd():

    ret = seed
    seed = (seed * 7 + 13) mod 1000000007
    return ret

for i = 1 to n:

    a[i] = (rnd() mod vmax) + 1

for i = 1 to m:

    op = (rnd() mod 4) + 1
    l = (rnd() mod n) + 1
    r = (rnd() mod n) + 1

    if (l > r):
         swap(l, r)

    if (op == 3):
        x = (rnd() mod (r - l + 1)) + 1
```

```
    else:
        x = (rnd() mod vmax) + 1

    if (op == 4):
        y = (rnd() mod vmax) + 1
```

Here op is the type of the operation mentioned in the legend.

Output

For each operation of types 3 or 4, output a line containing the answer.

Examples

input
10 10 7 9
output
2 1 0 3

input
10 10 9 9
output
1 1 3 3

Note

In the first example, the initial array is $\{8, 9, 7, 2, 3, 1, 5, 6, 4, 8\}$.

The operations are:

- 2 6 7 9
- 1 3 10 8
- 4 4 6 2 4
- 1 4 5 8
- 2 1 7 1
- 4 7 9 4 4
- 1 2 7 9
- 4 5 8 1 1
- 2 5 7 5
- 4 3 10 8 5