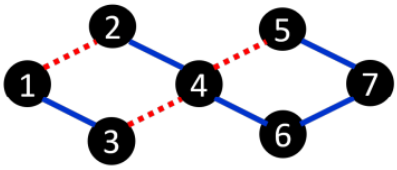
Problem A
A Coloring Game

Time Limit: 1 second
Memory Limit: 512 megabytes

Nhat and Thuc are playing a very simple game described as follows:



- Nhat tries to color all edges on a graph, which has N nodes and M undirected edges. Each edge can be blue or red.
- After Nhat colored the graph, Thuc tries to find a path that begins from node 1 and ends in node N, which has the minimum number of **bad points**. When Thuc moves on his chosen path from a red edge to a blue one (or vice versa), he gets one **bad point**.
- Nhat wants to color the graph in such a way that maximizes the bad points Thuc will get, regardless of which path Thuc will choose.

Could you determine the number of bad points Thuc will get if both players play optimally.

Note: A path is a sequence of edges where each pair of consecutive edges have a node in common.

Input

The first line contains two integers N and M.

In the next M lines, line i^{th} contains two integers u_i, v_i indicating an undirected edge between nodes u_i and v_i.

There is at most one edge between any pair of nodes.

Constraints:

- $2 \leq \mathrm{N}, \mathrm{M} \leq 10^5$,
- $1 \leq \mathrm{u_i}, \mathrm{v_i} \leq \mathrm{N}, \mathrm{u_i} \neq v_i$

Output

A single integer which is the number of bad points Thuc will get, assuming that both players play optimally.

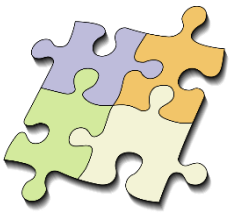
Sample Input	Sample Output
3 3 1 3 1 2 2 3	0

7 8 1 2 1 3 2 4 3 4 4 5 4 6 5 7 6 7	3

Problem B
Puzzles of a Puzzle

Time Limit: 10 seconds
Memory Limit: 512 megabytes

An has a complete puzzle of size $W \times H$. An breaks it down into N smaller puzzles, each has size $w_i \times h_i$. However, An does not remember correctly the height H of the complete puzzle, he only remembers its width W.



Please help An find the height H of the complete puzzle.

Input

The first line contains an integer W – the width of the complete puzzle.

The second line contains an integer N – the number of smaller puzzles.

On the next N lines, line i^{th} contains two integers w_i and h_i, which is the size of a small puzzle.

Constraints:

- $1 \leq \mathrm{N} \leq 5 \times 10^6$,
- $1 \leq \mathrm{W},\ \mathrm{H} \leq 10^4$,
- For each $1 \leq i \leq N, 1 \leq w_i, h_i \leq 10^4$.

Output

The output should contains an integer H, which is the height of the complete puzzle.

Sample Input	Sample Output
4 7 2 3 1 4 1 2 1 2 2 2 2 2 2 1	6

Problem C
Prime and Primechild

Time Limit: 2 seconds

Memory Limit: 512 megabytes

In this problem, you are playing with prime numbers. A prime number is defined as a natural number greater than 1 and not a product of two smaller natural number. An integer x is called **primechild** if it is a substring of a prime number.

An integer x is a substring of an integer y if it is equal to an integer derived from y by deleting zero or more consecutive digits of the most and least significant digits of y. For example, 124 is a substring of 124, 34124, 124983, 387812487, 124871248.

Task: You are given two integers l and h, along with a primechild p. Your task is to count the number of primes between the prime l^{th} and prime h^{th}, that contains the primechild p. For instance, for $l = 1, h = 10$, and $p = 9$, there are two primes 19 and 29 in between the 1^{st} prime 2 and the 10^{th} prime 29 that contain primechild 9.

Please note that the primechild p may include leading zeros, and a prime should be counted only once in case p occurs more than one time as its substring.

Input

The first line contains two integers l and h.

The second line contains the primechild p, which may have leading zeroes.

Constraints:

- $1 \leq l, h \leq 10^5$
- p is consisting of 1 to 6 digits.

Output

The output contains a single integers, which is the number of primes in the give range that contains the primechild p.

Sample Input	Sample Output
1 10 9	2
1 1000 00	10
500 1000 43	26

Problem D
XOR

Time Limit: 1 seconds

Memory Limit: 512 megabytes

XOR (eXclusive OR) is a Boolean logic operation that compares two input bits and generates one output bit. The logic is simple. If the bits are the same, the result is 0. If the bits are different, the result is 1. To compute the XOR of two integers, they are first converted to binary representations, then their bits are XOR-ed in order.

The Chinese magicians were looking to craft an item that consisted of a mixture of 2 of the N magic ingredients. However, the magicians were unable to perfectly fuse these two materials, resulting in the strength of the finished product being equal to the strength of material A XOR with the strength of material B.

From these N materials, magicians can craft $N \times (N-1)/2$ different products. The magicians wanted to know the strength of the weakest K of these variants.

Input

The first line contains two integers N and K.

In the next N lines, each line consist of an integer A_i indicating the strength of the i^{th} material.

Constraints:

- $N <= 10^5$
- $K <= min(25000, N \times (N-1)/2)$
- $A_i < 2^{31}$

Output

The output contains K numbers in ascending order indicating the strength of the weakest K products.

Sample Input	Sample Output
4 5 1 1 3 4	0 2 2 5 5

Explanation

In the sample input we have 4 numbers: $1, 1, 3, 4$. Therefore, there are $(4 \times 3)/2 = 6$ pairwise XOR's. These XOR's are:

$$1\ xor\ 1 = 0\ (A_1\ xor\ A_2)$$
$$1\ xor\ 3 = 2\ (A_1\ xor\ A_3)$$
$$1\ xor\ 4 = 5\ (A_1\ xor\ A_4)$$
$$1\ xor\ 3 = 2\ (A_2\ xor\ A_3)$$
$$1\ xor\ 4 = 5\ (A_2\ xor\ A_4)$$
$$3\ xor\ 4 = 7\ (A_3\ xor\ A_4)$$

If we sort these numbers we will obtain the sequence: $0, 2, 2, 5, 5, 7$. The first 5 numbers are: $0, 2, 2, 5, 5$.

Problem E
Binary Strings

Time Limit: 1 seconds

Memory Limit: 512 megabytes

Binh has N binary strings S of length M. The teacher asks Binh to choose a set of binary strings and keep the original order so that two consecutive strings do not have the same value at the same position ($S_{i,j} \neq S_{i+1,j}$).

Count the number of ways to choose a valid set. Since the number of possible choices is very large, print the number of possible choices modulo $10^9 + 7$.

Input

The first line contains T - the number of test cases.

For each test case,

- The first line contains 2 numbers N and M, which are the number of strings and length of the strings, respectively.
- The next N lines contain strings consisting only of 0 and 1.

Constrains:

- $T \leq 10$
- $N \leq 10\text{^}5$
- $M \leq 16$
- The sum of N over test cases does not exceed 5×10^5

Output

The output contains a single integer – The number of choices modulo $10^9 + 7$.

Sample Input	Sample Output
1 4 2 10 01 11 10	7

Problem F
Contiguous Sum

Time Limit: 2 seconds

Memory Limit: 512 megabytes

You are given a sequence A of n integers $a_1, a_2, \dots, a_n$ and m queries of type (u_i, v_i, k_i). For each query (u_i, v_i, k_i), your task is to find the largest contiguous sum of the integers in A from index u_i to v_i such that the sum is not exceed the value k_i.

Input

The first line contains n, m – the number of the integers in A and the number of queries, respectively.
The second line contains n integers $a_1, a_2, \dots, a_n$.
In the next m line, the i^{th} line contains three integers u_i, v_i, k_i, which represents a query.
Constrains:

- $1 \le n \le 2000$
- $1 \le m \le 2 \times 10^5$
- $-10^9 \le a_i \le 10^9$
- $1 \le u_i \le v_i, \le n$
- $-10^4 \le k_i \le 10^{14}$

Output

The output contains exacly m lines. The i^{th} line should contain the larges sum of contiguous integers in A from index u_i to v_i, that does not exeed k_i. If it is not possible to find such contiguous sum, the i^{th} line should contain "NONE".

Sample Input	Sample Output
5 3 1 -2 -3 5 4 1 3 -2 1 5 8 1 5 3	-2 6 2
6 4 3 8 -3 2 5 2 1 6 17 1 6 16 2 5 4 2 5 -4	17 15 4 NONE

Problem G
Dictionary

Time Limit: 1 seconds

Memory Limit: 512 megabytes

Bom is studying words in a language. He realizes that there are similar looking characters that may cause confusion for readers. Bom considers the "distance" between characters on a scale from 1 to 5, with lower being more similar. The distance between two same characters is 1. The "distinctiveness" of a word is defined as the sum of the distances between consecutive characters in a word.

For example, presume that the distance between "e" and "l" is 3, "l" and "y" is 2, "i" and "l" is 1. Then, the distinctiveness of the word "elly" is 3 + 1 + 2 = 6 (note that the distance between "l" and "l" is 1). The word "lily" has a distinctiveness value of 4, while "i" has a value of 0.

Given Bom's list of distances between characters and an integer N, Bom wants you to count the number of non-empty words with distinctiveness value at most N.

Input

The first line of input contains two integers, N ($1 \le N \le 10^9$) and M, the length of Bom's list of distances. Any distance not mentioned in the list is assumed to be 1.

For the next M lines, each line contains two characters L_1 and L_2 and an integer F, which means the distance between L_1 and L_2 is F. The distance between L_2 and L_1 is the same as the distance between L_1 and L_2. Every pair of characters will appear at most once.

Output

The output consists of a single integer, number of words with value at most N, modulo $10^9 + 7$.

Sample Input	Sample Output
20 10 e l 3 e o 1 o n 2 o r 4 r a 4 i n 5 e n 2 n t 3 t w 3 w i 5	470059518

Explaination

Examples of satisfying words: “elleonora”, “entwine”, “aaaaaaaaaaa”.

Problem H
Permutation

Time Limit: 1 second

Memory Limit: 512 megabytes

A permutation of n elements is an arrangement of integers in the range {1..n}. We list permutations in increasing lexicographic order and index them starting from 1. For example, when $n = 5$, we have:

1	1 2 3 4 5
2	1 2 3 5 4
3	1 3 2 4 5
…	…
120	5 4 3 2 1

Given n and a permutation $a_1, a_2, \ldots, a_n$. You must process q queries, each query is a pair (i, j) $(1 \le i \le j \le n)$ where you have to swap the elements at position i and j. After each query, you must print the index of the new permutation.

NOTE: the queries don't affect each other! In other words, the permutation is not changed after each query.

Input

The first line of input contains two integers, n $(1 \le n \le 3 \times 10^5)$ and q $(1 \le q \le 10^5)$.

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

For the next q lines, each line contains two integers i and j denoting a query.

Output

You should output q lines, each line an integer denoting the index of the new permutation modulo $10^9 + 7$, in the same order as the input.

Sample Input	Sample Output
5 3 1 5 4 2 3 1 3 2 3 2 5	91 17 9

Problem I
The IDs

Time Limit: 1 second

Memory Limit: 256 megabytes

Cuoi works as a manager in a company where employees are assigned an ID consisting of 4 alphanumeric characters. He thinks that some IDs are too similar, therefore one employee might mistakenly input an ID of another employee and the system will not recognize. Cuoi wants to address this problem, but before that, he has to obtain some statistics.

Given the list of IDs and an integer *D*, you need to find the number of pairs of IDs that differ in exactly *D* positions.

Input

The first line of input contains two integers, N ($2 \le N \le 50000$) and D ($1 \le D \le 4$), the number of employees and the number of different positions respectively.

For the next N lines, each line contains an ID. An ID is made of 4 alphanumeric characters, which is either a character in the English alphabet from "a" to "z", or a digit from 0 to 9. No two IDs will be the same.

Output

The output consists of a single integer, number of words with value at most N, modulo $10^9 + 7$.

Sample Input	Sample Output
4 1 0000 a010 0202 a0e2	0
4 2 0000 a010 0202 a0e2	3

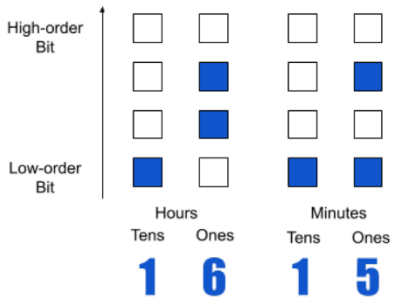
Explanation

In the first example, no two IDs differ in exactly 1 position. In the second example, there are 3 pairs that differ in 2 positions, those are: (0000, a010), (0000, 0202), and (a010, a0e2).

Problem J
Binary Clock

Time Limit: 2 seconds

Memory Limit: 512 megabytes



Tan has just bought a new clock for his room. The clock is a very special one and is readable by those who are familiar with binary numbers. The clock has four columns, each column representing one digit of the current time in hours and minutes.

Time is displayed in 24-hour format, with the 1st (left-most) column displaying the tens position for hours, the 2nd column displaying the ones position for hours, the 3rd column displaying the tens position for minutes, and the last (right-most) column displaying the ones position for minutes. For example, the time 16:55 would be displayed as shown in the figure.

Task: Write a program that takes a 24-hour time and print the corresponding clock face.

Input

The input contains only one line with a number of 4 digits describing a valid 24-hour time, between 0000 and 2359.

Output

The output contains four lines with a representation of the clock face displaying the given time. The tens of hours shall be in the 1st column, the single hours in the 3rd, the tens of minutes in the 7th, and the single minutes in the 9th. Use asterisks to represent bits that are set and periods to represent bits that are clear. Columns not used are to be filled with spaces. No extra whitespace are to appear at the beginning or end of any output line.

Sample Input	Sample Output
1615	. . . . . * . * . * . . * . * *
1900	. * . . . . * * . .
0830	. * . . . . * . . . * .

Problem K
Exam Grading

Time Limit: 1 seconds

Memory Limit: 64 megabytes

Professor T is grading the final exam of his students. The exam is in True/False format and each question is worth one point. Professor T wants to make the grades of his students look as good as possible, so he modifies the answers so that the lowest grade in his class is as high as possible.

Your task is to help professor T find the highest possible lowest grade that he can achieve.

Input

The first line contains two integers n and k, where n is the number of students, and k is the number of True/False questions on the exam.

Each of the next n lines contains a string of length k, consisting only of upper-case 'T' (for True) and upper case 'F' (for False). This string indicates the answers that a student has submitted, in the order the questions were given.

Constraints:

- $1 \leq n \leq 1000$
- $1 \leq k \leq 10$

Output

Output contains a single integer, which is the best possible lowest grade in the class.

Sample Input	Sample Output
5 4 TFTF TFFF TFTT TFFT TFTF	2
3 5 TFTFT TFTFT TFTFT	5

Problem L
Counting Trees

Time Limit: 1 seconds

Memory Limit: 512 megabytes

There are trees on a road. Each tree has a distinct height. Bill wants to count the number of unordered pairs of tree in on the road such that they are taller than every tree in between them.

More formally, let d be a sequence of the heights of the trees in order from left to right. Bill wants to count the number of pairs of indices i and j with $i < j$ such that for all k with $i < k < j$, $d_i > d_k$, and $d_j > d_k$. Note that if $j = i + 1$ (when there are no k's between i and j), it is true.

Input

The first line contains an integer n, which is the number of trees on the road.

Each of the next n line contains a single integer d_i. These are the heights of the trees, in the order of where they are on the road. The sequence is guaranteed to be permutation of the integers 1 through n.

Output

The output contains a single integer, which is the number of pairs of trees which are taller than every tree in between them.

Sample Input	Sample Output
3 2 1 3	3

Problem M
A Cute Number

Time Limit: 1 seconds

Memory Limit: 512 megabytes

A cute number is a natural number that has no leading zeroes and has no two adjacent digits the same.

You task is to find the number of cute numbers in between lower bound L and upper bound U (inclusive).

Input

The first line contains a single integer L, which is the lower bound.

The second line of the input contains a single integer U, which is the upper bound.

Constraints:

- $1 \leq L \leq U < 10^{10^5}$
- Note that the limits are not a misprint; L and U can be up to 10^5 digits.

Output

Output a single integer, which is the number of rainbow numbers between L and U (inclusive). Because this number may be very large, output it modulo 998244353.

Sample Input	Sample Output
1 10	10
12345 65432	35882