Problem Miko Wants to Survey

Input data	stdin
Output data	stdout

As a member of the Public Morals Committee, Miko needs to ensure all of Shuchi'in Academy's students abide by the school rules. However, as diligent as she may be, she cannot oversee each and every student's behaviour. Or can she?...

First, Miko sets up K surveillance cameras numbered from 1 to K, the i^{th} camera being placed at coordinates (X_i, Y_i) in Shuchi'in Academy's garden. All cameras can be set to survey the same radius R: if the distance between a point S in the garden and at least one of the cameras is R or smaller, then point S is *surveyed*.

Next, Miko directs the flow of students going to classes as follows: they enter the garden through point A at coordinates (X_A, Y_A), follow a path P that Miko will designate and then go into the Academy's main building through point B at coordinates (X_B, Y_B). The path P will need to be a polygonal chain (set of linked line segments) starting in point A and ending in point B. Naturally, every point of P will also need to be surveyed.

Miko is now wondering what is the minimum radius R to set the cameras, such that there exists a path P between points A and B which is completely surveyed, and has a total length of at most L. *(I want to watch the students as closely as possible, but I don't want to make them late to classes!)*

Input Data

The first line contains two space-separated integers K and L, with the respective meanings from the statement above. The second line contains four space-separated integers X_A, Y_A X_B and Y_B giving the coordinates of points A and B. Then K lines follow: the i^{th} of these contains two space-separated integers X_i and Y_i, the coordinates of the i^{th} surveillance camera.

Output Data

Output a single line containing a floating point number R, the answer to Miko's question.

Restrictions

- $K \leq 15$
- The coordinates of points A, B and each camera are integers between $-1\,000$ and $1\,000$.
- $L \leq 1\,000\,000\,000$
- L is larger or equal to the distance between points A and B.
- Your output R will score full points for a test if its absolute or relative difference to the correct answer R_c is at most 10^{-4}, that is, if $\frac{|R-R_c|}{max(1,|R_c|)} \leq 10^{-4}$.

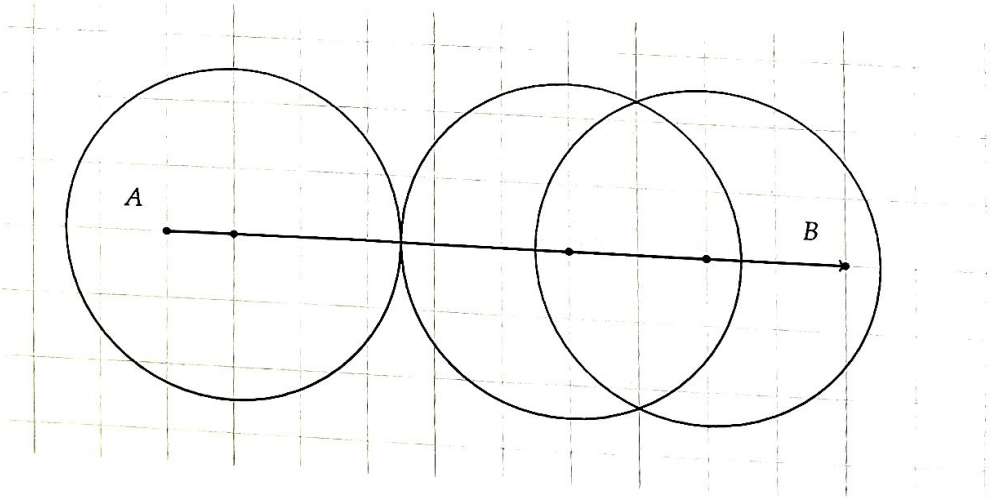
#	Points	Restrictions
1	7	Points A, B and all cameras have the Y coordinate equal to 0.
2	10	$L = 1\,000\,000\,000$
3	24	L is equal to the straight line distance between A and B
4	28	$K = 3$, $(X_1, Y_1) = (X_A, Y_A)$ and $(X_2, Y_2) = (X_B, Y_B)$
5	31	No further restrictions.

Examples

Input data	Output data
3 20 0 0 10 0 1 0 6 0 8 0	2.500000
3 12 0 0 10 0 0 0 5 7 10 0	4.460034

Explanations for example 1. We have $K = 3$ surveillance cameras and, after setting the minimum possible radius R, we need to have a path P from point $A = (0,0)$ to point $B = (10,0)$ that is completely surveyed and has length of at most $L = 20$.

This minimum possible radius is $R = 2.5$, and a path P we can choose is the straight line segment between $A = (0,0)$ and $B = (10,0)$, which has length 10. Camera 1 surveys all points lying on the segment between $(0,0)$ and $(3.5,0)$, camera 2 surveys all points lying on the segment between $(3.5,0)$ and $(8.5,0)$, and camera 3 surveys all points lying on the segment between $(5.5,0)$ and $(10,0)$.



Problem KDAG

Input data	stdin
Output data	stdout

You are given a directed graph with N vertices and M edges and an integer K. You need to partition the M edges of the graph into K disjoint subsets such that the graph formed by the edges from the subset is a directed acyclic graph. If there is more than one partition, determine a partition where the difference between the size of the biggest subset and the size of the smallest subset is minimised.

Interaction protocol

You must implement a function:

```
vector<int> partition(int N, int M, int K, vector<int> U, vector<int> V);
```

The first and second parameters represent two given integers N, M, the number of the vertices and the number of edges of the graph. The third parameter is K, the number of subsets. The fourth and fifth parameters represent two arrays U and V containing M values, each between 1 and N. These arrays represent that for every i from 0 to $M-1$ there is a directed edge from the vertex U_i to vertex V_i. The function must return an array P of M integers with values between 1 and K. P_i is the index of the subset in which the i^{th} edge was partitioned into. If such a partition does not exist then return an empty array.

The contestant should not implement the `main` function. This will be implemented in the committee's `grader.cpp` file; you will be given a sample `grader.cpp` in the attachments. Our `main` function will read N, M, K and M pairs of integers, representing an edge between two vertices, and output `partition(N, M, K, U, V)`. This is the input format used in the example below.

Restrictions

- $2 \leq N, M \leq 1\,000\,000$
- $2 \leq K \leq M$

#	Points	Restrictions
1	4	$K = 2$ and $N, M \leq 1\,000$
2	9	$N, M \leq 1\,000$
3	23	$K = 2$ and $N \leq 2\,000$
4	25	$N \leq 2\,000$
5	20	$K = 2$
6	19	No further restrictions.

Examples

Input data	Output data
4 4 3 1 2 2 4 1 3 3 4	1 1 2 3

Problem Revstring

Input data	`stdin`
Output data	`stdout`

Nano the robot loves embroidery. In front of her, she has a piece of cloth formed out of N square units, each embroidered with a different pattern. Each pattern is codified by a letter of the Latin alphabet (a, ..., z), and the pattern of the i-th square from left to right is S_i.

Nano can now modify the piece of cloth in the following way: she cuts out a continuous sequence of square units, then puts them back in place in reverse order. For instance, if she modifies the underlined characters in this way, the operation has the following effect:

$$\text{abcabc} \rightarrow \text{abbacc}.$$

Nano's friend the professor now asks her a question: how many different pieces of cloth could be reached by applying exactly one modification to the original piece of cloth? For some unknown reason, the professor is only interested in the remainder of the answer when divided by $10^9 + 7$.

Input data

The input contains two lines. The first line contains the number N. The second line contains the sequence S, written as a string of characters.

Output

The output should contain the answer to the professor's question, modulo $10^9 + 7$.

Restrictions

- $N \leq 1\,000\,000$

#	Points	Restrictions
1	4	$S =$ aaaaa...
2	6	$S =$ ababab..., N is even
3	7	$S =$ abcabcabc..., N is divisible by 3
4	14	$N \leq 100$
5	17	$N \leq 2\,000$
6	25	$N \leq 100\,000$
7	27	No further restrictions.

Examples

Input data	Output data	Explanations
4 xabx	6	The possible results are xabx, axbx, baxx, xbax, xxba, xaxb.
10 abdsgahwql	45	

Problem Hunter X Hunter

Input data	stdin
Output data	stdout

Our hero Gon landed on a island called Greed Island. Greed Island has exactly N towers placed one next to another on a straight line. Gon noticed that these are not ordinary towers, they are magical and he can tell you the following facts about them:

- The towers are indexed with numbers from 1 to N.
- You are given the height H_i for each tower, which is the height of the i^{th} tower at moment 0.
- The towers are growing at a constant rate R_i per time unit. At moment t, the height of the i^{th} tower will be $H_i + t * R_i$.
- When Gon visits the i^{th} tower *for the first time* he is granted C_i coins.

Gon can travel according to the following rules:

- He can only move from a tower to a neighbouring tower. That is, from the tower with index i he can only move to the towers with index $i-1$ (if $i > 1$) or index $i+1$ (if $i < N$).
- He can always go from a tower to a shorter neighbouring tower.
- Gon can also move from a tower to a taller neighbour one **only if** the height difference between the two towers is smaller or equal than a constant value K.
- Every move he makes from one tower to another takes exactly one unit of time.
- The rules for moving between towers are considered for the tower heights at the time on which he starts moving between them.
- Gon can choose to stay in a tower as many whole units of time as he wants. The towers will continue growing either way.

Gon is very curious about how he can travel on Greed Island so he wants you to answer Q queries. For each query you are given the index of two towers A and B. Gon wonders what is the maximum number of coins he can collect on a journey that starts from tower A at time 0 and ends in tower B. Gon can take as much time as he wants for each journey.

Input data

The first line contains three integers N, K, Q. The following N lines contain three integers describing the N towers: H_i, R_i, C_i. The following Q lines contains two integers A and B, representing the queries you need to answer.

Output data

The output consists of Q lines, the i^{th} line containing the answer for the i^{th} query. If Gon cannot reach tower B starting from A, the answer is -1.

Restrictions

- $1 \leq K \leq 10^{12}$
- $1 \leq H_i \leq 10^{12}$
- $1 \leq R_i \leq 1\,000\,000$
- $1 \leq C_i \leq 10^9$

#	Points	Restrictions
1	6	$N, Q \leq 100$, $K \leq 10\,000$, $H_i \leq 10\,000$, $R_i \leq 1\,000$
2	7	$N, Q \leq 100$
3	14	$N \leq 1\,000$, $Q \leq 100$
4	17	$N, Q \leq 2\,500$
5	37	$Q \leq 50$, $1 \leq N \leq 10\,000$
6	19	$Q \leq 100$, $1 \leq N \leq 100\,000$

Examples

Input data	Output data
5 3 2 1 3 4 10 1 2 7 2 5 3 5 3 6 4 8 2 4 1 3	18 11

Problem ISQ

Input data stdin
Output data stdout

You are given two arrays A and B of length N which are 1-indexed, and an integer M. You should answer Q queries of the form: *Given (a, b, c, d), what is the maximum value of $(A_i + B_j) \mod M$, where $a \leq i \leq b$ and $c \leq j \leq d$?*

Input data

The first line of the input contains the integers N, M and Q. The next two lines contain the arrays A, and B respectively. The next Q lines contain four integers, each line representing a query.

Output

The output should contain Q lines, the i^{th} line containing the answer to the i^{th} query.

Restrictions

- $N, Q \leq 80\,000$.
- $M \leq 10^9$
- $A_i, B_i < M$ for every $1 \leq i \leq N$
- $a \leq b$ and $c \leq d$

#	Points	Restrictions
1	6	$N, Q \leq 500$
2	10	$N, Q \leq 10\,000$
3	5	$Q \leq 10000$, $b - a \leq 5\,000$ and $d - c \leq 5\,000$
4	8	$Q \leq 10000$, $b - a \leq 10\,000$ and $d - c \leq 10\,000$
5	8	$b - a \leq 100$
6	35	$Q \leq 15\,000$
7	28	No additional restrictions

Examples

Input data	Output data
4 10 4 2 5 9 1 1 2 8 5 1 2 2 2 2 4 1 1 1 1 1 4 1 4 2 4	7 6 7 9

Problem Peaks

Input data stdin
Output data stdout

You are given a matrix A with N rows and M columns. Each column contains a permutation of the numbers from 1 to N. A column j is said to form a peak, if there exists a row index k such that $A_{1,j} < A_{2,j} < \ldots < A_{k,j} > A_{k+1,j} > \ldots > A_{N,j}$. For instance, $[1,2,3,4]$, $[1,2,4,3]$ and $[1]$ are peaks, while $[3,1,2]$ and $[1,3,2,4]$ are not. You are allowed to remove 0, 1, or more rows from the matrix. However, you can not remove all rows from A. Count how many ways X there are to remove rows from the matrix such that each column of the remaining matrix forms a peak, modulo $10^9 + 7$.

Input data

The first line of the input contains the integers N and M. The next N lines contain M integers each, representing the matrix A.

Output

The output should contain an integer, representing the number X modulo $10^9 + 7$.

Restrictions

- $1 \leq N, M \leq 500$

#	Points	Restrictions
1	5	$N, M \leq 15$
2	23	$M \leq 10$, $N \leq 100$
3	51	$N, M \leq 100$
4	21	No further restrictions.

Examples

Input data	Output data	Explanations
4 3 1 4 3 4 2 1 2 3 4 3 1 2	11	When removing the second row the matrix becomes 1 4 3 2 3 4 3 1 2 All columns now form a peak: $1 < 2 < 3$ $4 > 3 > 1$ $3 < 4 > 2$