Problem A. Alice, Bob and Rectangles

Input file:	*standard input*
Output file:	*standard output*
Time limit:	3 seconds
Memory limit:	256 mebibytes

There is a sequence of N rectangles $\{R_1, R_2, \ldots, R_N\}$, where $R_i = \{(x, y) : X_{i1} \le x \le X_{i2}, Y_{i1} \le y \le Y_{i2}\}$.

Firstly, Alice picks an non-empty subsequence of $\{R_1, R_2, \ldots, R_N\}$, namely $\{R_{k_1}, R_{k_2}, \ldots, R_{k_n}\}$, where $1 \le k_1 < k_2 < \cdots < k_n \le N$.

After that, Bob chooses a sequence of *lattice points* $\{p_1, p_2, \ldots, p_{n+1}\}$ which:

- $p_1 \in R_{k_1}$
- $\forall 2 \le i \le n, p_i \in R_{k_{i-1}} \cap R_{k_i}$
- $p_{n+1} \in R_{k_n}$

Work out the number of distinct pairs $(\{k_1, k_2, \ldots, k_n\}, \{p_1, p_2, \ldots, p_{n+1}\})$.

Since the answer may be very large, you should output the result modulo $(10^9 + 7)$ instead.

Input

The first line contains an integer N. The next N lines contain 4 integers $X_{i1}, Y_{i1}, X_{i2}, Y_{i2}$.

$(1 \le N \le 5 \cdot 10^4, |X_{i1}|, |Y_{i1}|, |X_{i2}|, |Y_{i2}| \le 10^9, X_{i1} \le X_{i2}, Y_{i1} \le Y_{i2})$

Output

An integer denotes the number.

Example

standard input	standard output
2 1 1 2 2 2 2 3 3	48

Note

$(\{1\}, \{(2, 2), (2, 2)\})$ and $(\{2\}, \{(2, 2), (2, 2)\})$ are two different ways.

Problem B. Bugs

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	256 mebibytes

N bugs move on the plane. At time t, coordinates of i-th bug can be calculated by the formula $(X_i + A_i \cdot t, Y_i + B_i \cdot t)$.
Find $\frac{\int_0^T S(t)\mathrm{d}t}{T}$ for given T, where $S(t)$ is the area of convex hull of the N points (at time t).

Input

The first line contains 2 integers N, T. The next N lines contain 4 integers X_i, Y_i, A_i, B_i.
$(1 \le N \le 50, 1 \le T \le 10^4, |X_i|, |Y_i|, |A_i|, |B_i| \le 50)$

Output

A floating number denotes the desired value. Absolute or relative error within 10^{-9} is accepted.

Example

standard input	standard output
3 1 0 0 0 0 0 1 0 0 1 0 1 0	0.750000

Note

$S(t) = \frac{1+t}{2} \implies \frac{\int_0^T \frac{1+t}{2}\mathrm{d}t}{T} = \frac{3}{4}$

Problem C. Counting

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	256 mebibytes

Given integers N and M and K points with coordinates (X_i, Y_i). Your task is to calculate the number of (x_{min}, x_{max}, y_{min}, y_{max}) such as:

- $x_{min}, x_{max}, y_{min}, y_{max} \in \mathbb{N}$
- $1 \le x_{min} \le x_{max} \le N$
- $1 \le y_{min} \le y_{max} \le M$
- $\exists k : (x_{min} \le X_k \le x_{max}) \wedge (y_{min} \le Y_k \le y_{max})$ ($\wedge$ for logical and)

Since the answer may be very large, print it modulo $(10^9 + 7)$.

Input

The first line contains of the input contains three integers N, M, and K. Each of the next K lines contain two integers X_i, Y_i.

$(1 \le N, M \le 10^9, 0 \le K \le 1000, 1 \le X_i \le N, 1 \le Y_i \le M)$

Output

Print one integer — the answer to the problem modulo $10^9 + 7$.

Example

standard input	standard output
2 2 1 1 1	4

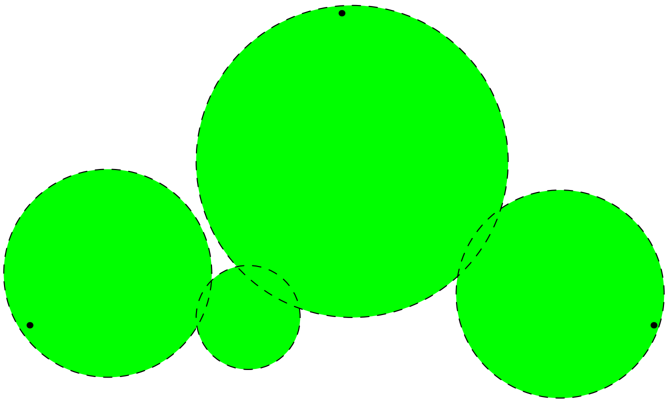
Problem D. Dracula's Garden

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	256 mebibytes

Count Dracula is a voracious (as well as carnivorous) gardener. He keeps a small garden within easy walking distance of his eponymous mansion, and naturally also near to his eponymous potting shed.

Recently, the Count has had to cut back his gardening time sharply on account of adversely excellent weather conditions. His prize Deadly Nightshades are starting to feel neglected.

He would like to pave a shaded path between the garden, the mansion, and the shed using an ensemble of variously-sized trees he's obtained from Harker Nurseries at suspiciously low prices. Each tree casts a circle-shaped shadow that Dracula can walk across.



Is it possible for him to re-plant his trees in such a way that they cast a continuous shadow on the ground connecting the three given locations? You may assume shadows are directly beneath the trees and do not move throughout the day.

Input

The input consists of:

- three lines each containing the integer co-ordinates of x_i and y_i, each the planar coordinates of one gardening location where ($-10^6 \le x_i, y_i \le 10^6$ for each i).
- one line with an integer n, ($1 \le n \le 12$): the number of trees the Count has acquired.
- one line containing n integers $s_1 \ldots s_n$ ($1 \le s_i \le 10^6$ for each i), the individual radii of the shadows cast by the Count's tree collection.

Output

If the Count can place the trees in such a way as to connect all of the sites, output "`possible`". Otherwise, output "`impossible`".

Examples

standard input	standard output
-6 0 6 0 0 6 4 1 3 1 1	impossible
-6 0 6 0 0 6 4 2 3 2 1	possible

Problem E. Editor

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	256 mebibytes

You are going to implement the most powerful editor for integer sequences.

The sequence is empty when the editor is initialized.

There are 5 types of instructions.

`I` x Insert x after the cursor.

`D` Delete the element before the cursor.

`L` Move the cursor left unless it has already been at the begin.

`R` Move the cursor right unless it has already been at the end.

`Q` k Suppose that the current sequence **BEFORE the cursor** is $\{a_1, a_2, \ldots, a_n\}$. Find $\max_{1 \le i \le k} S_i$ where $S_i = a_1 + a_2 + \cdots + a_i$.

Input

The first line of the input contains an integer Q, which is the number of instructions. Each of the next Q lines contain an instruction as described above.

($1 \le Q \le 10^6$, $|x| \le 10^3$ for `I` instruction, $1 \le k \le n$ for `Q` instruction)

Output

For each instruction of type Q, print the answer to query.

Example

standard input	standard output
8 I 2 I -1 I 1 Q 3 L D R Q 2	2 3

Note

The following diagram shows the status of sequence after each instruction:

```
I 2  [2]
       ^
I -1 [2, -1]
           ^
I 1  [2, -1, 1]
              ^
Q 3  [2, -1, 1]
              ^
L    [2, -1, 1]
           ^
D    [2, 1]
       ^
R    [2, 1]
          ^
Q 2  [2, 1]
          ^
```

Problem F. Flow

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	256 mebibytes

For undirected graph G, let $F_{i,j}$ be the maximum flow running from node i to node j. ($F_{i,i} = -1$ for convenience)

Determine whether the graph G exists for given $F_{i,j}$.

Input

The first line of the input contains an integer N — the number of nodes of graph G. The next N lines contain N integers $F_{i,j}$.

$(1 \leq N \leq 100, F_{i,i} = -1, \forall i \neq j, F_{i,j} = F_{j,i}, 0 \leq F_{i,j} \leq 10^9)$

Output

Print "YES" if graph G exists, or "NO" otherwise.

If the answer is YES, print $G_{i,j}$. $(G_{i,i} = -1, \forall i \neq j, G_{i,j} = G_{j,i}, 0 \leq G_{i,j} \leq 10^9)$

Examples

standard input	standard output
3 -1 2 2 2 -1 2 2 2 -1	YES -1 1 1 1 -1 1 1 1 -1
3 -1 1 2 1 -1 2 2 2 -1	NO

Problem G. Group

Input file:	*standard input*
Output file:	*standard output*
Time limit:	10 seconds
Memory limit:	256 mebibytes

Let Sym_N be the set of all permutations of $\{1, 2, 3, \ldots, N\}$. For given $S \subseteq \mathrm{Sym}_N$, find $\min |G|$ which:

- $S \subseteq G$
- $\forall \sigma, \tau \in G, \sigma \circ \tau^{-1} \in G$

Remark: For $\sigma = (\sigma(1), \sigma(2), \ldots, \sigma(N)), \tau = (\tau(1), \tau(2), \ldots, \tau(N))$, $\sigma \circ \tau = (\sigma(\tau(1)), \sigma(\tau(2)), \ldots, \sigma(\tau(N)))$

Input

The first line contains 2 integers N, M, where $|S| = M$. The next M lines contain N integers $\sigma_i(1), \sigma_i(2), \ldots, \sigma_i(N)$. $(1 \le N \le 50, 1 \le M \le 100)$

Output

An integer denotes $\min |G|$.

Examples

standard input
3 2 2 1 3 3 2 1

standard output
6

standard input
50 2 2 1 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 1

standard output
30414093201713378043612608166064768844377641568960512000000000000

Note

The new lines in the second sample are made for clearness.

Problem H. How Much Y-Triples

Input file: *standard input*
Output file: *standard output*
Time limit: 1 second
Memory limit: 256 mebibytes

For given tree T, count the number of sets $\{A, B, C\}$ such as:

- A, B, C are nodes of tree T
- There exists **NO** simple path which covers A, B, C

Input

The first line contains an integer N, which is the size of tree T. The next $(N-1)$ lines contain 2 integers A_i, B_i, which denotes the edge (A_i, B_i).

Note that the nodes are numbered by $1, 2, \ldots, N$.

$(3 \le N \le 10^5, 1 \le A_i, B_i \le N)$

Output

An integer denotes the number.

Example

standard input	standard output
4 1 2 1 3 1 4	1

Note

The only set is $\{2, 3, 4\}$.

Problem I. Integer Sum

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	256 mebibytes

For given N, let $S(k)$ be the number of $(x_1, x_2, \ldots, x_k)$ which:

- $x_1, x_2, \ldots, x_k \in \mathbb{Z}^+$
- $x_1 + x_2 + \cdots + x_k = N$

Find $(S(1) + S(2) + \cdots + S(N)) \bmod (10^9 + 7)$.

Input

The first line contains an integer N.
$(1 \le N < 10^{100000})$

Output

An integer denotes the value.

Example

standard input	standard output
2	2

Note

For $N = 2$, $S(1) = S(2) = 1$.

Problem J. Japanese Olympics Center

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	256 mebibytes

The ICPC training camp is held in an accommodation facility called Japanese Olympics Center. Today is the last day of the camp. Unfortunately, you are ordered to check if all the participants have properly cleaned their rooms.

The accommodation facility is a rectangular field whose height is H and width is W. The field is divided into square cells. The rows are numbered 1 through H from top to bottom and the columns are numbered 1 through W from left to right. The cell in row i, column j is denoted by (i, j). Two cells are adjacent if they share an edge.

Each cell is called either a wall cell or a floor cell. A wall cell represents a wall which no one can enter. A floor cell is a part of the inside of the accommodation facility. Everybody can move between two adjacent floor cells. The floor cells are divided into several units. Each unit is assigned an uppercase English letter (A, B, C, etc). A floor cell adjacent to exactly one floor cell is called a room. Otherwise the floor cell is called an aisle. For example, the accommodation facility can be shown as the following figure. We denote a wall cell by '`.`' (single period).

```
...................
.....AAABBBBBBB....
...A.AA.A...B.B..B.
..AAAAAAAABBBBBBBB.
...A..A.A.....B....
......A.......BBBB.
....A.AA..C.C...B..
...AAAAACCCCCCBBBB.
...A..A...C.C...B..
...................
```

In the above figure, there are 7 rooms in unit A, 4 rooms in unit B, and 4 rooms in unit C.

Because the accommodation facility is too large to explore alone, you asked the other participants of the camp to check the rooms. For simplicity's sake, we call them staffs. Now, there are K staffs at the cell (s, t). You decided to check the rooms according to the following procedure.

1. First, you assign each staff to some of units. Every unit must be assigned to exactly one staff. Note that it is allowed to assign all the units to one staff or to assign no units to some staffs.

2. Then, each staff starts to check the rooms in the assigned units at the same time. The staffs can move between two adjacent floor cells in T_{move} time. To check the room at (i, j), the staffs must move to the cell (i, j) and spend T_{check} time there. Each staff first determines the order of the units to check and he or she must check the rooms according to the order. For example, suppose that there is a staff who is assigned units A, C, and E. He may decide that the order is $E \to A \to C$. After that, he must check all the rooms in unit E at first. Then, he must check all the rooms in unit A, and so on. The staffs can pass any floor cells. However, the staffs cannot check rooms that are not assigned to them. Further, the staffs cannot check rooms against the order of units that they have decided.

3. After checking all the assigned rooms, the staffs must return to the cell (s, t).

4. When every staff returns to the cell (s, t), the task is done.

Because you do not have enough time before the next contest, you want to minimize the total time to check all the rooms.

Input

The first line of the input contains three integers H, W, and K ($1 \le H \le 50$, $1 \le W \le 50$, $1 \le K \le 12$). The second line contains four integers s, t, T_{move}, and T_{check} ($1 \le s \le H$, $1 \le t \le W$, $1 \le T_{move}, T_{check} \le 10^4$). Each of the following H lines contains exactly W characters. Each character represents a cell in the accommodation facility. The j-th character in the i-th line of these lines is '`.`' if the cell (i, j) is a wall cell. Otherwise, the character is an uppercase English letter ('`A`'-'`L`') representing the unit to which the cell (i, j) belongs.

You can assume that the following conditions are satisfied.

- The number of rooms in each unit is between 1 and 12, inclusive.
- The cell (s, t) is guaranteed to be an aisle.
- Floor cells in each unit are connected.
- Floor cells in the field are connected.
- Every unit contains at least two cells.

Output

Output the minimum required time to check all the rooms in one line.

Examples

standard input	standard output
3 3 1 1 1 10 10 AAA A.. A..	100
3 3 2 1 1 10 10 ABB A.. A..	50

standard input	standard output
5 10 3 3 6 1 100 ...G.H.A.. .AAGAHAABB FFAAAAAAA. .EEAAADACC ..E...D...	316
10 19 2 6 15 3 10 ....AAABBBBBBB.... ...A.AA.A...B.B..B. ..AAAAAAAABBBBBBBB. ...A..A.A.....B.... ......A.......BBBB. ....A.AA..C.C...B.. ...AAAAACCCCCCBBBB. ...A..A...C.C...B..	232
27 36 6 24 19 616 1933 ....B...............B......... ..........BBB..........BBBB......... ..........BBBBB.......BBBB.......... ...........BBBBBBBBBBBBBBB.......... ...........BBBBBBBBBBBBBBB.......... ...........BBBBBBBBBBBBBB........... ............BBBBBBBBBBBBB........... ...........BBBBBBBBBBBBBBB.......... ..........BBBBBBBBBBBBBBBBB......... ......B...BBBBBBBBBBBBBBBBB...B..... ......BB.BBBBBBBBBBBBBBBBBB..BB..... .......BBBBBBBBBBBBBBBBBBBB.BB...... .........BBBBBBBBBBBBBBBBBBBB....... ........BBBBB..BBBBBBBB..BBBBB...... ......BBBBBB....BBBBB....BBBBBB..... .....BBBBBBBB...BBBBB..BBBBBBBB.B... ...BBBBBBB..B...BBBBB..B..BBBBBBBB.. .BBBBBBBBB.....BBBBBBB......BBBBBBB. ..BBBBBBB......BBBBBBB........BBB... .BBBB.........BBBBBBBBB........BBB.. ....BBBBBBBBB............. ....BBBBBBBBBB............. ....BBBBBBBBBB............. ....BBBBBBBB.............. ....BBBBBBBB..............	137071