Problem A. Array

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

Ranran has a sequence a of n integers $a_1, a_2, \cdots, a_n$ which satisfies $\sum \frac{1}{a_i} \le \frac{1}{2}$ and he is very proud of it, so he comes up with a problem for you.

You need to find out a sequence c of m integers. With c, you construct an infinite sequence b, and b_i equals to $c_{i \bmod m}$. b must satisfy the condition that in every consecutive a_i numbers of b there exists a number equals to i.

Please note that a is 1-indexed and b, c are 0-indexed. The value of m is decided by you.

Can you solve the problem?

Input

The first line contains an integer $n (1 \le n \le 10^5)$.

The second line contains n integers $a_1, a_2, \cdots, a_n (2 \le a_i \le 2 \times 10^5, \sum \frac{1}{a_i} \le \frac{1}{2})$.

Output

The first line output an integer m.

The second line output m integers $c_0, c_1, \cdots, c_{m-1}$.

You should guarantee that $1 \le m \le 10^6$ and $1 \le c_i \le n$.

Example

standard input	standard output
1 2	2 1 1

Problem B. Eezie and Pie

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	256 megabytes

Eezie, a pie maniac, would like to have some pies with her friends on a hot summer day. However, the weather is so hot that she can't go outdoors and has to call for the delivery service.

The city Eezie lives in can be represented by N nodes connected by $N-1$ edges, and the city center is node 1. In other words, the city is a rooted tree, root of which is node 1. There are N pie houses in the city, the i-th on node i. For some reason, a pie house on node i can only deliver its pie to nodes on the simple path from node i to node 1.

Eezie is a bit worried that a pie might lose its flavor during the deliver. After careful calculation, she decided that a pie from the i-th pie house can maintain its flavor if the distance it is delivered does not exceed its flavor-loss-distance d_i. The distance between two nodes on the tree is the number of **edges** on the simple path between them.

Now, Eezie wants to order some pies for all her friends who live on different nodes of the tree. Therefore, she wants you to calculate for each node how many pie houses can deliver their pies to the node without flavor loss.

Input

The first line contains an integer $N(1 \le N \le 2 \times 10^6)$, representing the number of nodes of the city Eezie lives in.

The second line contains N integers $d_1, d_2, \cdots, d_N (1 \le d_i \le N)$, representing the maximum travel distances for pies from pie houses.

Each of the next $N-1$ lines contains two integers $u, v (1 \le u, v \le N)$, representing an edge. It is guaranteed that the edges form a tree.

Output

Output N integers in a line, the i-th integer representing the answer to node i.

Example

standard input	standard output
10 1 2 2 3 2 4 3 5 4 6 4 7 1 8 8 9 8 10 0 0 1 2 2 5 3 1 0 2	6 6 2 3 1 1 1 2 1 1

Problem C. Forest

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	512 megabytes

Dense forest in the Western suburbs helps you break loose.

Given a weighted undirected simple graph with n vertices and m positive-weighted edges. Sum up the weight of the minimum spanning forest for each of its spanning subgraph, and print the value modulo 998244353 as answer.

Input

The first line consists of a single integer $n (n \leq 16)$, denoting the number of vertices in the graph.

The next n lines describes the graph with an adjacency matrix, more specifically:

- Each of the n lines consists of n non-negative numbers.
- The j-th number of the i-th line $A_{i,j} (0 \leq A_{i,j} \leq 10^9)$ is 0, if there's no edge between vertex i and j; or the weight of the only edge between them otherwise.

It's guaranteed that $A_{i,i} = 0$ and $A_{i,j} = A_{j,i}$ for each $1 \leq i, j \leq n$, which ensures the graph is an undirected simple graph with all edges positive-weighted. It's also guaranteed that the number of edges m does not exceed 100.

Output

Print a single integer in the first line, which is the value modulo 998244353.

Example

standard input	standard output
4 0 1 1 1 1 0 1 1 1 1 0 1 1 1 1 0	158

Note

A *minimum spanning forest* of a weighted directed graph G with vertex set V and edge set E is defined as follows:

- It's a subset of E (not necessarily non-empty or different from E), denoted as S.
- Any pair of vertices (u, v) that can reach each other in G can still reach each other only using edges in S.
- S is the subset with the minumum weight among all subsets satisfying the above two conditions. The weight of S is the sum of weight of all edges in it.

A *spanning subgraph* of G is a graph with vertex set V and edge set a subset of E (not necessarily non-empty or different from E).

Problem D. Fourier and Theory for the Universe

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	128 megabytes

Note: the description is really long. For better understanding, important conceptions are in **bold** style.

Fourier has started a project with his friends, aiming to research into the ultimate theory for the whole universe.

Georg Cantor, the trailblazer in set theory, uses 'Marvelous Cantor Set(MCS)' to describe the universe. In plain language, the MCS **contains all the integers** from 1 to n.

René Descartes, who has invented a lot of important operators, uses 'Profound Descartes Operator(PCO)' to describe how the elements in MCS interact with each other. In plain language, when PCO is exerted on exactly two elements in MCS, the **product of the two elements** shall be returned. Moreover, the MCS is **only defined** between two numbers with product no more than n.

Leonhard Euler, the master of number theory, has discovered the fundamental elements in MCS. He names them 'Fabulous Euler Number(FEN)'. In his research, he finds that **all the PCOs of two different prime numbers**, are FEN.

George Boole, the pioneer in boolean logistic, has perfected the definition of FEN. He uses 'Excellent Boole Law(EBL)' for it, which contains two descriptions:

- **the PCO of two FENs are FEN**, as long as the PCO between them is defined.
- **no other numbers** than **all the PCOs of two different prime numbers and the PCOs of two FENs**, are FEN.

Through hundreds and thousands of great scientists' investigations, the algebra in MCS has finally be fulfilled. However, the law of converting of normal natural numbers to elements in MCS hasn't been found, and in long time periods, MCS algebra was not widely used.

Finally, Fourier, the genius transformation researcher, advanced 'Final Fourier Transformation(FFT)' for this problem. The theorem is an epochal discovery, although with a small flaw: FFT requires to **count the number of FENs in MCS**, however Fourier doesn't know how to do it!

Help Fourier with this problem! If you can solve it successfully, there may be a constant named after you!

Input

One line with a single number, $n (1 \le n \le 10^{11})$.

Output

One line with a single number, which is the number of FENs in MCS.

Example

standard input	standard output
10	2

Note

In this case, only 6 and 10 are FENs.

Problem E. From AtCoder

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

You have an $n \times n$ grid. Each cell contains an integer. The number at the i-th row and j-th column has initial value $a_{i,j}$.

In one operation you can choose a cell (i, j) and an integer x such that $|x| \le 10^9$ and do the following:

- Add x to each number in the i-th row. Then, subtract x from each number in the j-th column.

Determine whether you can make all numbers non-negative within 1000 operations. If so, construct a solution.

Input

The first line contains a single integer $n (1 \le n \le 501)$.

The next n lines each contains n integers. The j-th integer on the i-th line is $a_{i,j} (0 \le |a_{i,j}| \le 10^6)$.

Output

If your goal cannot be achieved, print one integer -1 in a line.

Otherwise print integer $k (0 \le k \le 1000)$ in the first line, indicating the number of operations. Then print k lines. Each line contains three integers i, j, x separated by a single space, indicating one operation. You must make sure $1 \le i, j \le n, 0 \le |x| \le 10^9$.

Examples

standard input	standard output
3 6 5 -4 -7 2 0 3 26 47	5 1 1 3 2 1 -12 2 2 10 3 2 -10 3 3 -2
3 -1 -2 -3 -4 -5 -6 -7 -8 -9	-1
3 0 1 2 3 4 5 6 7 8	0

Problem F. Hash

Input file:	`standard input`
Output file:	`standard output`
Time limit:	7 seconds
Memory limit:	256 megabytes

NIO has a tree T rooted at 1.

He defines the hash function of T as $F(T) = \left(\sum_{i=1}^{n}\sum_{j=i+1}^{n} X^iY^jZ^{\mathrm{lca}(i,j)}\right) \bmod P$, where $P = 998244353$ and $\mathrm{lca}(i,j)$ is the lowest common ancestor of i and j.

Unfortunately, he lost the tree in an accident. The only thing he remembers is $F(T)$.

Now given $F(T)$ and X, Y, Z, you need to reconstruct T with no more than 50 vertices.

Input

The first line contains an integer $T (1 \le T \le 20)$ indicating the number of test cases.

For each test case, the only line contains four integers $F(T), X, Y, Z (0 \le F(T) < P, 2 \le X, Y, Z \le P - 2)$.

It is guarenteed that X, Y, Z are chosen randomly from range $[2, P - 2]$.

Output

For each test case, output n lines, where n is the number of vertices of the tree.

The first line output n. You should make sure that $1 \le n \le 50$.

The next $n - 1$ lines output two integers u, v indicating an edge in your tree. These $n - 1$ edges should form a tree.

Example

standard input	standard output
1 36 2 3 2	2 1 2

Problem G. Icon Design

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

What's the feeling of designing an icon for a school as a programmer? Now you have a chance doing it!

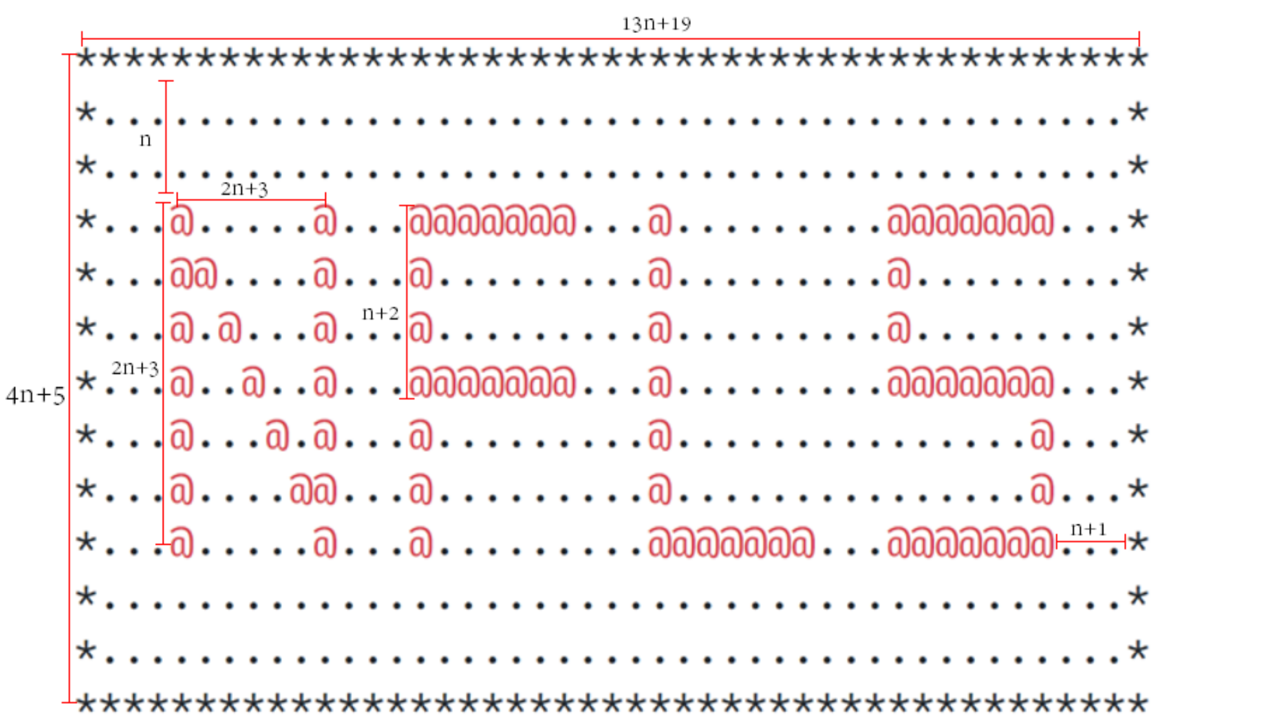
The icon of Nanjing Foreign Language School (NFLS for short) is not complicated, it can be represented as an ASCII art.

```
********************************
*..............................*
*..@...@..@@@@@..@......@@@@@..*
*..@@..@..@......@......@......*
*..@.@.@..@@@@@..@......@@@@@..*
*..@..@@..@......@..........@..*
*..@...@..@......@@@@@..@@@@@..*
*..............................*
********************************
```

Since the icon might be used in different places, you need to print the icon in different size.

Given size n, print the icon of size n.

Detailed format is shown below, you can also look at the sample output to confirm it.



Like which is shown in the picture, '*' is used on the boundary, and '@' is used for the letters.

$(n+1)$ '.'s are used to seperate letters and the boundary horizontally, and n '.'s are used vertically.

Each letter is $(2n+3)$ characters wide and $(2n+3)$ characters in height.

The icon's size is $(4n+5) \times (13n+19)$.

Input

The first line contains a positive integer $n (1 \le n \le 5)$, representing the size of the icon.

Output

Print the icon of size n.

Example

standard input	standard output

1	`********************************`
	`*..............................*`
	`*..@...@..@@@@@..@......@@@@@..*`
	`*..@@..@..@......@......@......*`
	`*..@.@.@..@@@@@..@......@@@@@..*`
	`*..@..@@..@......@..........@..*`
	`*..@...@..@@@@@..@@@@@..@@@@@..*`
	`*..............................*`
	`********************************`

Note

The output when $n = 3$ is so large that it is shown in the link below: https://paste.ubuntu.com/p/2vFVnhfpYQ/.

Problem H. Jumping Steps

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	512 megabytes

Liu Kanshan loves jumping steps!

Now in front of Liu there're n steps. He stands on the 0th step and wants to jump to the n-th step. He can only jump up (from the x-th step to the $x+k$-step, for any positive integer k). For one jump, he gets a score of k^2.

However, m of these steps are broken and Liu can't jump onto them. The broken steps are $p_1, p_2, \cdots, p_m$.

Also, if Liu jumps over more than S broken steps in one jump (that means $\sum_{i=1}^{m}[x < p_i < y] > S$, where x and y are the start position and the end position of this jump), he won't get the score for this jump because of overwork. **Note that the score he gets before won't be cleared.**

Now Liu wonders what is the sum of the total scores he can get in all different possible ways of jumping. Two ways of jumping are different if and only if there exists a step that it is jumped on in one way, and not jumped on in the other way. Since the answer may be very large, you need to find it modulo 10^9+7.

Input

The first line contains three integers n, m and $S (1 \le n \le 10^9, 0 \le S \le m \le 2 \times 10^6)$.

The second line contains m integers $p_1, p_2, \cdots, p_m (0 < p_1 < p_2 \cdots < p_m < n)$.

Output

The only line contains one integer: the answer modulo 10^9+7.

Examples

standard input	standard output
3 1 1 2	14
6 2 1 2 4	60
8 2 2 2 5	854

Note

Explanation of example #2:

Let q be the sequence of steps Liu jumps on. There are 8 ways of jumping:

1. $q = \{0, 1, 3, 5, 6\}$, the total score is $1^2 + 2^2 + 2^2 + 1^2 = 10$.
2. $q = \{0, 1, 3, 6\}$, the total score is $1^2 + 2^2 + 3^2 = 14$.
3. $q = \{0, 1, 5, 6\}$, the total score is $1^2 + 0 + 1^2 = 2$.
4. $q = \{0, 1, 6\}$, the total score is $1^2 + 0 = 1$.
5. $q = \{0, 3, 5, 6\}$, the total score is $3^2 + 2^2 + 1^2 = 14$.
6. $q = \{0, 3, 6\}$, the total score is $3^2 + 3^2 = 18$.

7. $q = \{0, 5, 6\}$, the total score is $0 + 1^2 = 1$.
8. $q = \{0, 6\}$, the total score is 0.

The sum of them is $10 + 14 + 2 + 1 + 14 + 18 + 1 + 0 = 60$.

Problem I. Line

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Ranran has a set s_v of n vectors and an integer d. He is bored at Sunday so he decides to invent a new problem for you.

You need to give a set s_p of points, size of which is m. You will pick up every point (a_i, b_i) in s_p and every vector (x_j, y_j) in s_v. The pair (a_i, b_i, x_j, y_j) is called good if and only if the line visits exactly d points in s_p. s_p is good if and only if every pair satisfies the condition. You need to find out a good set of points.

Ranran thinks about it at a few sleepless nights thinking of Yangyang and solves it. Now he gives this problem to you. Can you solve it?

Input

The first contains two integers n and $d(1 \le n, d \le 6)$.

Each of the next n lines contains two integers $x_i, y_i (0 \le x_i, y_i \le 6, x_i + y_i > 0)$.

Attention, it is not guarenteed that all vectors are pairwise different.

Output

Output an integer m in the first line.

Each of the next m lines, output two integers a_j, b_j.

You should guarantee that $1 \le m \le 10^5$, a_j, b_j are 32-bit signed integers and points are pairwise different.

Example

standard input	standard output
1 1 1 0	1 0 0

Note

The conditions can be satisfied using only one point.

Problem J. Number Game

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

There are three integers A, B and C written on the blackboard.

You can perform the following two operations as many times as you like:

- Change B to $A - B$.
- Change C to $B - C$.

Please note that each time you don't need to perform all two operations. You can choose one type of operation to perform.

You are given an integer x. Answer whether you can change C into x using these operations.

You need to answer T queries independently.

Input

The first line contains a positive integer $T (1 \le T \le 10^5)$.

Each of the next T lines contains four integers $A, B, C, x (-10^8 \le A, B, C, x \le 10^8)$.

Output

For each test case, output 'Yes' if C can become x, and 'No' otherwise (without quotes).

Example

standard input	standard output
3 2 4 3 1 2 4 3 2 4 2 2 0	Yes No Yes

Note

Please note that A, B, C, x could be negative.

Problem K. SolarPea and Inversion

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	512 megabytes

SolarPea thinks inversion is beautiful.

For a 01-sequence Z with length n and a constant c, SolarPea defines the rating of Z is $T(Z,c) = c^{\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}[Z_i>Z_j]}$

PolarSea has two integer sequences X and Y with length $k(\forall 1 \le i \le k, 1 \le X_i \le n, 0 \le Y_i \le 1)$. PolarSea likes a 01-sequence Z with length n if and only if $\forall 1 \le i \le k, Z_{X_i} = Y_i$.

SolarPea wrote all 01-sequences which have length n and contain m '1's on the paper. PolarSea saw it and crossed out all sequences that he doesn't like. Now you're given c, please calculate the sum of the ratings of the remaining sequences on the paper.

Since the answer could be very large, you should output it modulo 1065977431 (a prime number).

It is guaranteed that c is generated randomly.

Input

The first line contains four non-negative integers n, m, k, c

$(1 \le n \le 10^{18}, 0 \le m \le \min(n, 10^7), 0 \le k \le \min(n, 30), 2 \le c < 1065977431)$.

The next k lines, each line contains two non-negative intergers X_i and $Y_i (1 \le X_i \le n, 0 \le Y_i \le 1)$.

It is guaranteed that all X_i are pairwise different.

Output

Output the answer modulo 1065977431.

Examples

standard input	standard output
3 2 1 10 2 1	101
4 2 1 10 2 1	10110
1004535809 115194 2 21658 822 1 1064 0	606261277

Note

Explaination of sample #1:

There are two remaining sequences $\{1,1,0\}$ and $\{0,1,1\}$. $T(\{1,1,0\},c) = c^2 = 100$, $T(\{0,1,1\},c) = c^0 = 1$, so the answer is $100 + 1 = 101$.

Problem L. Striking String Problem

Input file:	`standard input`
Output file:	`standard output`
Time limit:	8 seconds
Memory limit:	1024 megabytes

NIO is given two strings S and T consisting of lowercase letters, an integer k and $2k$ integers $l_i, r_i (1 \le i \le k)$.

Define $U = S[l_1, r_1] + S[l_2, r_2] + \cdots + S[l_k, r_k]$. He has q queries, each described by two integers x and y. For a query, he wants to know the number of times that T appears in $U[x, y]$. Help him!

- $S[l, r]$ is $S_l S_{l+1} \cdots S_r$.
- $S + T$ is $S_1 S_2 \cdots S_{|S|} T_1 T_2 \cdots T_{|T|}$.

Input

The first line contains a single string $S (1 \le |S| \le 10^6)$.

The second line contains a single string $T (1 \le |T| \le 5 \times 10^5)$.

The third line contains two integers k and $q (1 \le k, q \le 5 \times 10^5)$.

Each of the next k lines contains two integers l_i and $r_i (1 \le l_i \le r_i \le |S|)$.

Each of the next q lines contains two integers x_i and $y_i (1 \le x_i \le y_i \le |U|)$.

Output

Print q lines. The i-th of them contains a single integer - the answer to the i-th query.

Example

standard input	standard output
abaaba abaa 8 10 1 6 4 6 3 6 2 4 5 6 2 2 5 5 1 4 1 24 21 24 1 15 1 23 6 6 4 7 1 8 8 16 16 23 7 20	5 1 3 4 0 1 2 1 0 2

Note

$U =$ `abaabaabaaababaababbabaa`.

Problem M. Z-Game on Grid

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Alice and Bob are playing a game on an $n \times m$ grid where each cell has either 'A', 'B' or '.' written on it. They take turns moving a chess piece on the grid and Alice moves first.

Initially the piece is on cell $(1, 1)$. In each player's turn, he or she can move the piece one cell right or one cell down. That is, if the piece is on cell (x, y) before the turn, the player can move it to $(x + 1, y)$ or $(x, y + 1)$, as long as it doesn't go beyond the grid.

At any time, if the piece is on a cell with 'A', Alice wins and the game ends. If the piece is on a cell with 'B', Bob wins and the game ends. If the piece reaches cell (n, m) without the game ending, then it is a draw.

Since Alice cannot decide what acts Bob will take, she would like to know if she can be in control of the situation. Given the grid they're playing on, can you tell her if she can always find a way to win, draw or lose the game no matter what acts Bob takes?

Input

In the first line an integer $T (1 \le T \le 50)$, representing the number of test cases.

For each test case, the first line contains two integers $N, M (1 \le N, M \le 500)$, representing the grid's size.

Each of the next N lines for the case contains M characters (either 'A', 'B' or '.'), describing the grid.

Output

For each test case, output three words 'yes' or 'no' in one line, representing if Alice can find a way to win, draw or lose the game respectively (without quotes).

Example

standard input	standard output
2 3 3 ..B ..B BB. 1 3 ...	no no yes no yes no