Problem A. Moniphant Sleep

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	1024 megabytes

Hello, here is Moniphant! He has a problem for you to solve.

Moniphant discovered that he can traverse multiple levels of sleep, encountering a Mofunfun in each one.

Formally, there are 4 types of operations.

1. **Sleep**: Moniphant descends to the next level of sleep.
2. **Wake up**: Moniphant ascends from the current level of sleep, i.e., he goes to the previous level.
3. **Say "You are a pig!" to Mofunfun**: This insults the Mofunfun at the current level, making it angry.
4. **Mofunfun's retaliation**: Mofunfuns at every level attempt to retaliate against Moniphant, but only the angry ones succeed. Successful retaliation sends Moniphant back to the level of the least infuriated Mofunfun, i.e., the smallest level where a Mofunfun is angry. After retaliating, the Mofunfun is no longer angry. Note that if there are no angry Mofunfuns, this operation is omitted.

The four operations correspond to the pictures above.

Note that Mofunfun of the current level disappears when Moniphant goes back to smaller levels.

Now consider a sequence of n Moniphants, indexed from 1 to n. Your task is to perform one of the 4 operations on all Moniphants indexed between l and r, or determine the sleeping level of a specific Moniphant.

Each Moniphant undergoes the "Sleep" operation (the 1st operation) 5×10^5 times before the problem commences.

Input

The first line of the input contains two integers n and q ($1 \le n, q \le 5 \times 10^5$), indicating the number of Moniphant and the number of operations.

In the following q lines, there are three integers op, l, r ($op \in \{1, 2, 3, 4, 5\}, 1 \le l \le r \le n$), if $op \le 4$, it means you should do that type of operation for Moniphants in range $[l, r]$. And if $op = 5$, then $l = r$ holds, and you should print the sleeping level of the l-th Moniphant.

Output

For every operation 5, output a single line contains a single integer, indicating the ansewr.

Examples

standard input	standard output
1 9 1 1 1 1 1 1 1 1 1 3 1 1 2 1 1 1 1 1 1 1 1 4 1 1 5 1 1	500004
3 7 2 1 3 3 1 3 1 1 3 1 1 3 5 1 1 4 1 3 5 1 1	500001 499999

Note

At last, Moniphant hopes you enjoy the contest!

Problem B. Click the Circle

Input file: `standard input`
Output file: `standard output`
Time limit: 5 seconds
Memory limit: 1024 megabytes

This is a modified version of the game osu!.

There are two types of objects:

1. Circle: c_x, c_y, t:
   - The center of this circle is at (c_x, c_y), and its hit time is t.
   - The circle's presence in the game is during the time interval $[t - d, t + d]$.
   - All circles share the same radius, r.

2. Slide: s_x, s_y, t_x, t_y, u, v:

   A slide contains two parts:

   (a) A moving circle, which also has a radius of r.

   (b) A frame that holds the path of this circle.

   A slide holds the path of a moving circle, whose radius is always r.

   The move can be described as follow:

   (a) At time $u - d$, the circle appears with the frame, and the center of the circle is (s_x, s_y)

   (b) At time u, the center of the circle starts moving towards (t_x, t_y) at a constant speed.

   (c) At time v, the center of the circle reaches (t_x, t_y).

   (d) After time $v + d$, the circle and the frame disappear.

Two objects are considered to intersect if, at a certain time t, both are present and their shapes overlap (boundaries inclusive).

The two components of a slide should be treated as **two** distinct objects.

Given the values r, d, and n objects, calculate the number of intersecting pairs of objects.

Input

The first line of the input contains there positive integers n, r, d $(1 \le n, r, d \le 10^3)$.

The next n lines describes the objects. Each line starts with a number $type \in \{1, 2\}$, indicating the type of the object.

- If $type = 1$, then there will be three integers c_x, c_y, t, indicating a circle.
- If $type = 2$, then there will be six integers s_x, s_y, t_x, t_y, u, v, indicating a slide.

It is guaranteed that $1 \le c_x, c_y, s_x, s_y, t_x, t_y \le 10^4$, $1 \le t, u, v \le 10^3$ and $u < v$.

Output

Output a single line contains a single integer, indicating the number of pairs.

Examples

standard input	standard output
2 1 1 1 1 1 2 1 2 2 3	1
2 1 1 1 1 1 2 1 3 2 3	0
2 1 1 1 3 3 2 2 5 5 5 1 2 4	3
2 1 1 2 1 1 1 5 2 4 2 5 5 5 1 2 4	2
2 1 1 2 10 1 10 20 2 4 2 1 10 20 10 2 4	6

Problem C. Tree

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	1024 megabytes

You are given a tree with n vertices. The i-th vertex has a color denoted by a_i, and the i-th edge connects the fa_i-th vertex with the $(i+1)$-th vertex. This edge has a color represented by fc_i and a length indicated by fw_i.

A simple path is defined as good if and only if all vertices on the path share the same color and all edges along the path also share a common color. Note that the color of the vertices and the color of the edges can be different.

There are q operations to be performed. In the i-th operation, the color of the vertex a_{x_i} is changed to c_i.

At the beginning, and after each operation, you need to determine the maximum length of a good path.

Input

The first line of the input contain two positive integers n, q $(1 \le n, q \le 2 \times 10^5)$.

The next line contains n integers $a_1, \cdots, a_n$ $(1 \le a_i \le n)$.

The next line contains $n-1$ integers $fa_2, \cdots, fa_n$ $(1 \le fa_i < i)$.

The next line contains $n-1$ integers $fc_2, \cdots, fc_n$ $(1 \le fc_i \le n)$.

The next line contains $n-1$ integers $fw_2, \cdots, fw_n$ $(0 \le fw_i \le 10^9)$.

The i-th of the next q lines contains two integers x_i and c_i $(1 \le x_i, c_i \le n)$.

Output

You need to output $q+1$ lines.

The first line of the output contains a single integer, indicating the maximum length of a good path before all the queries.

Then, for each query, output a single line contains a single integer, indicating the maximum length of a good path after the query.

Example

standard input	standard output
5 5 5 4 3 4 5 1 2 3 1 2 2 2 2 4 9 2 6 2 5 4 5 5 4 3 5 2 1	6 10 10 4 15 2

Problem D. Alice and Bob

Input file: `standard input`
Output file: `standard output`
Time limit: 1 second
Memory limit: 1024 megabytes

Alice and Bob are playing a game.

They are given a permutation p, and take turns to perform the following operation, with Alice going first:

- Operation: Rearrange $p_{1 \cdots p_1}$ in any desired order.

If someone do two operations with the same p_1, he or she loses.

Alice and Bob are both strategically adept and will always choose the optimal operation to secure a win.

Given all permutations of size n, determine how many of them Bob will win, modulo 998244353.

Input

The first line of the input contains a single integer n ($1 \le n \le 10^7$).

Output

Output a single line contains a single integer, indicating the answer.

Examples

standard input	standard output
1	1
2	1
10	997920
100	188898954

Problem E. Neighbourhood

Input file:	standard input
Output file:	standard output
Time limit:	10 seconds
Memory limit:	1024 megabytes

You are given a tree with n vertices. Each edge has a weight w_i.

There are q operations of the following two types:

- 1 i c: Change w_i to c.
- 2 x d: Count number of $y(1 \le y \le n)$ such that the shortest path between x and y is not greater than d.

Input

The first line of the input contains two integers n and q $(2 \le n \le 2 \times 10^5, 1 \le q \le 2 \times 10^5)$.

The next $n-1$ lines, each line contains three integers x_i, y_i, w_i $(1 \le x_i, y_i \le n, 1 \le w_i \le 10^9)$, representing an edge connecting x_i and y_i with weight w_i.

The next q lines, each line contains three integers 1 i c $(1 \le i \le n-1, 1 \le c \le 10^9)$ or 2 x d $(1 \le x \le n, 0 \le d \le 2 \times 10^{14})$, indicating an operation.

Output

For each operation of type 2, print one line with a single integer, indicating the answer.

Example

standard input	standard output
3 7 1 2 3 2 3 1 2 2 1 2 1 3 2 3 4 1 1 1 2 2 1 2 1 0 2 3 1	2 2 3 3 1 2

Problem F. Cover

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	1024 megabytes

You are given a tree with n vertices and $n-1$ edges. The degree of each vertex is at most k.

There are m undirected simple paths; the i-th path starts at vertex a_i, ends at vertex b_i, and carries a weight of w_i. We say an edge e is covered by a path (x, y) if and only if vertices x and y are disconnected when we remove edge e.

Please find a subset S of these paths such that each edge is covered by S at most once. Your goal is to maximize $\sum_{i \in S} w_i$.

Input

The first line of the input contains three integers n, m, k ($2 \le n \le 10^5, 0 \le m \le 5 \times 10^5, 1 \le k \le 12$).

The next $n-1$ lines, each line contains two integers x, y ($1 \le x, y \le n$), denoting there is an edge connecting vertex x and y in the given tree.

The i-th of the next m lines contains three integers a_i, b_i, w_i ($1 \le a_i, b_i \le n, 0 \le w_i \le 10^9, a_i \ne b_i$).

Output

Output a single line contains a single integer, indicating the answer.

Example

standard input	standard output
5 7 3 1 2 1 3 2 4 2 5 3 2 8 5 4 10 3 1 2 1 2 7 2 1 2 1 2 1 5 2 3	19

Problem G. Teleport

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	1024 megabytes

You are now piloting a UFO in an $n \times n$ grid formed by points (x, y) where $1 \le x, y \le n$. Some points are impassable (*) and others are passable (.).

Initially, you are at point $(1, 1)$, and you aim to reach (n, n) as quickly as possible. When you are at point (x, y), you can teleport to $(x+1, y)$, $(x, y+1)$, $(x-1, y)$, $(x, y-1)$, or $f^i(x, y)$ for any non-negative integer $i \le k$ in one second. The function $f^i(x, y)$ is defined as:

$$f^i(x, y) = \begin{cases} (x, y) & (i = 0) \\ f^{i-1}(y+1, x) & (i > 0) \end{cases}$$

You cannot teleport if the target location is outside the grid or if the target location is impassable.

Find the minimum time required to reach (n, n). If you can never reach (n, n), print `-1`.

Input

The first line of the input contains two integers n and k $(1 \le n, k \le 5000)$.

Each of the next n lines contains n characters, representing the grid.

It is guranteed that points $(1, 1)$ and (n, n) are passable.

Output

One integer in a line representing the minimum time to reach (n, n), or `-1` if it is unreachable.

Examples

standard input	standard output
3 2 .*. .*. ...	3
3 3 .*. .*. ...	2

Problem H. Function

Input file:	standard input
Output file:	standard output
Time limit:	1 second
Memory limit:	1024 megabytes

Ranran have a function f , and now he wants to find the value of $f(1) \bmod 998244353$

It is known that:

$$f(x) = \begin{cases} 0 & x > n \\ 1 + \sum_{k=2}^{20210926} f(kx) & \end{cases}$$

Input

The first line of the input contains a single integer n ($1 \le n \le 10^9$).

Output

Output a single line contains a single integer, indicating the ansewr.

Examples

standard input	standard output
1	1
2	2
100	949

Problem I. Digit

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2.5 seconds
Memory limit:	1024 megabytes

Given a positive integer n, in each turn:

1. Uniformly choose a digit d from n (in decimal representation).
2. Update n by setting $n \leftarrow n \cdot (d + 1)$.

Calculate the expected number of turns it takes for n to exceed N, modulo 998244353.

Input

There are multiple test cases in a single test file.

The first line of the input contains a single integer T $(1 \le T \le 200)$, indicating the number of the test cases.

For each test case, the first line of the input contains two integers n and N $(1 \le n \le N \le 10^{18})$.

Output

For each test case, output a single line contains a single integer, indicating the answer.

It can be proved that the answer always exists.

Example

standard input	standard output
3 1 10 1 100 1 1000	3 4 942786340

Problem J. Leaves

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	64 megabytes

L has a binary tree with each leaf node u labeled a_u.

If we traverse the entire tree in order (left child, then right child), we can place all the leaf nodes in a sequence.

Now, L will perform the following operation **exactly** m times:

1. Choose a non-leaf vertex a.
2. Swap the left child and right child of vertex a.

After these operations, L wants you to determine the lexicographically minimum sequence that he can achieve.

Input

The first line of the input contains two integers n and m ($0 \le m \le \frac{n-1}{2}, n \le 1000, 2 \nmid n$).

Then n lines, the i-th line starts with an integer $type \in \{1, 2\}$.

- if $type = 1$, then two integers l_i, r_i ($i < l_i, r_i$), indicating the left and right child of i, respectively.
- if $type = 2$, then a single integer $a_i (1 \le a_i \le 10^9)$, indicating the label of this leaf.

Output

Output a line contains $\frac{n+1}{2}$ integers, indicating the optimal sequence.

Examples

standard input	standard output
3 0 1 2 3 2 1 2 2	1 2
7 1 1 2 3 1 4 5 1 6 7 2 4 2 2 2 3 2 1	2 4 3 1
7 2 1 2 3 1 4 5 1 6 7 2 4 2 2 2 3 2 1	1 3 4 2

Problem K. water235

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	1024 megabytes

Given an $N \times M$ matrix, your task is to fill all the cells with water, using the least amount of water possible.

According to the rules of `Minecraft`, if a cell is empty and at least two of its neighboring cells (a neighboring cell is a cell that shares an edge) are filled with water, then this cell will be filled with water.

The water-filling process continues until there are no more empty cells adjacent to at least two cells with water.

Input

The first line of the input contains two integers N and M ($1 \leq N \times M \leq 10^6$), indicating the size of the matrix.

Output

The first line of the output contains a single integer, indicating the minimum number of 1.

The next N lines contain a 0/1 matrix of $N \times M$. In the matrix, 1 represents you will fill the water in this grid initially, and 0 means empty.

If there exists multiple solutions, you may print any of them.

Examples

standard input	standard output

`2 1`	`2`
	`1`
	`1`

`3 3`	`3`
	`1 0 1`
	`0 0 0`
	`0 1 0`

Problem L. Square

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	1024 megabytes

Suppose you have a positive integer x, you can transform it into $x - 1$ or $x + \lfloor \sqrt{2x} + 1.5 \rfloor$ in a single operation.

Find the minimum number of operations required to transform it into another positive integer y.

Input

There multiple test cases in a single test file.

The first line of the input contains a single integer T $(1 \le T \le 10^5)$, indicating the number of the test cases.

For each test case, the first line of the input contains two integers x_i and y_i $(1 \le x_i, y_i \le 10^{18})$.

Output

For each test case, output a single line contains a single integer, indicating the answer.

Example

standard input	standard output
2 5 1 1 5	4 3

Problem M. Delete the Tree

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	1024 megabytes

You are given a graph with n vertices numbered from 1 to n and $n-1$ edges. It is guaranteed that at the beginning, every pair of points can be reached from each other, meaning this graph is actually a tree. You are such a tree-hater so you decide to delete this tree!

However, you are only allowed to perform the following operation: select some currently existing vertices. These selected vertices must not be connected by any edges. For each selected vertex v, add an edge between any pair current neighbours of v (a vertex u is a neighbour of v if and only if there is an edge between u and v), and then delete the selected vertex and all its incident edges (an edge e is incident to v if and only if v is one of the endpoints of e). It can be proved that the order of performing the above operation on the selected vertices in each operation does not affect the result. Also note that the operation may introduce multiple edges in the graph.

You can perform this operation no more than 10 times. Please output a sequence of operations so that all vertices will be deleted after the operations. If there are multiple possible sequence of operations, output any will be considered correct. It can be proved that there is at least one feasible operation.

Input

The first line contains an integer n ($3 \le n \le 500$), denoting the number of vertices in the initial graph.

Then $n-1$ lines follow, with the i-th line ($1 \le i \le n-1$) containing two integer x_i, y_i ($1 \le x_i < y_i \le n$), denoting there is an edge connecting vertex x_i and y_i in the initial graph.

It is guaranteed that for any $1 \le i < j < n$, $x_i \ne x_j$ or $y_i \ne y_j$.

Output

The first line contains an integr m, denoting the number of operations. You should make sure that $0 \le m \le 10$.

Then m lines follow, with the i-th line describing the i-th operation: the first integer k contains the number of vertices selected in this operation, then k distinct integers in the same line separated by spaces denoting the label of vertices selected in this operation.

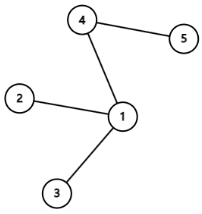
Example

standard input	standard output
5 1 2 1 3 1 4 4 5	4 2 1 5 1 2 1 4 1 3

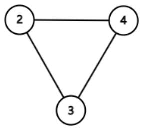
Note

The operations corresponding to the first example's output is shown below.

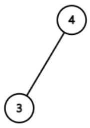
Before the first step, a picture of the graph is shown below:



After the first step, a picture of the graph is shown below:



After the second step, a picture of the graph is shown below:



After the third step, a picture of the graph is shown below: