

2026 Canadian Computing Olympiad
Day 1, Problem 1
Waterloo Tag

Time Limit: 4 seconds

Memory Limit: 512 MB

Roger and Troy are playing a game of tag at the University of Waterloo. The University of Waterloo can be represented as N buildings connected by M sidewalks. The i -th sidewalk connects buildings a_i and b_i , and is d_i metres long. There is at most 1 sidewalk between any pair of buildings. The sidewalks do not intersect (i.e. you can only move from one sidewalk to another at a building), and they might not lie on a plane (due to bridges and tunnels). Starting from any building, it is possible to reach any other building by walking along the sidewalks.

Roger starts the game of tag at building 1 and he can walk up to v_1 metres per second. Roger can also wait at a building or wait anywhere on a sidewalk. Roger will walk in a way that maximizes the duration of the game of tag.

Troy will pick a building x and release a group of students at building x . The students will spread out at v_2 metres per second along all sidewalks. The game of tag is over when Troy's students reach Roger.

For each building x , how long will the game of tag last?

Input Specification

The first line of input will consist of 4 space-separated integers N, M, v_1, v_2 ($2 \leq N \leq 2\,000$; $N - 1 \leq M \leq 5\,000$; $1 \leq v_1, v_2 \leq 100$).

The next M lines each contain 3 integers, where the i -th line contains integers a_i, b_i, d_i ($1 \leq a_i < b_i \leq N$; $1 \leq d_i \leq 10\,000$).

The following table shows how the 25 available marks are distributed:

Marks Awarded	Additional Constraints
3 marks	$N = 3$ and $M = 2$.
3 marks	$N = 3$ and $M = 3$.
7 marks	$v_1 = v_2 = 1$ and all sidewalks are 2 metres long ($d_i = 2$).
7 marks	$N \leq 100$ and $M \leq 200$.
5 marks	None.

Output Specification

Output $N - 1$ lines, where the i -th line contains the duration of the game of tag in seconds if Troy releases a group of students at building $i + 1$. You must output the duration as a fraction in simplest terms.

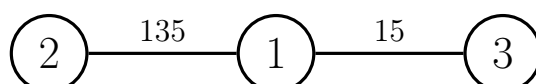
Note that an integer d is a *divisor* of an integer q if there is no remainder when q is divided by d . An integer z is a *common divisor* of integers x and y if z is a divisor of both x and y . A fraction x/y is in simplest terms if y is positive, and x and y do not have a common divisor greater than one.

Sample Input 1

```
3 2 1 10
1 2 135
1 3 15
```

Output for Sample Input 1

```
15/1
5/3
```

Explanation of Output for Sample Input 1

For $x = 2$, Roger should walk to building 3. After 15 seconds, the students tag Roger at building 3 and the game of tag is over.

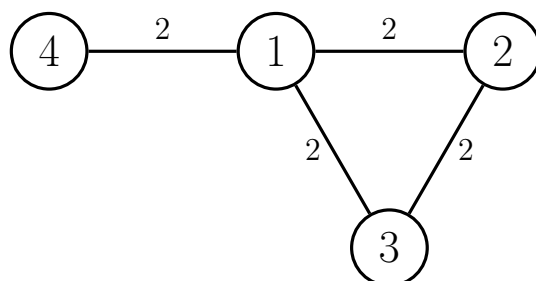
For $x = 3$, Roger should walk towards building 2. After $5/3$ seconds, the students tag Roger at the sidewalk between buildings 2 and 3 and the game of tag is over. Notice that Roger walked $1.666\dots$ metres and the students walked $15 + 1.666\dots$ metres.

Sample Input 2

```
4 4 1 1
1 2 2
1 3 2
2 3 2
1 4 2
```

Output for Sample Input 2

```
4/1
4/1
5/1
```

Explanation of Output for Sample Input 2

For $x = 2$, Roger should walk to building 4.

For $x = 3$, Roger should walk to building 4.

For $x = 4$, Roger should walk to the centre of the sidewalk between buildings 2 and 3.

2026 Canadian Computing Olympiad
Day 1, Problem 2
Melborp

Time Limit: 3 seconds

Memory Limit: 512 MB

Seta is creating problems for the CCO! She came up with the following problem:

Given an array $A[1, \dots, N]$ whose values are in the range $[1, N]$, define $B[i]$ to be the number of pairs (ℓ, r) such that $\ell \leq i \leq r$ and

$$A[i] = \min(A[\ell], A[\ell + 1], \dots, A[r - 1], A[r]).$$

Print the array $B[1, \dots, N]$.

However, the day before the CCO, Seta's computer crashed, and she was only able to recover the output files. Given the output array $B[1, \dots, N]$, can you write a program to reconstruct the input array $A[1, \dots, N]$?

Seta reminds you that the array A is not necessarily unique, and she will accept any valid array.

Input Specification

The first line of input will contain a single integer, N . The second line of input will contain N space-separated integers $B[1], \dots, B[N]$ ($1 \leq B[i] \leq N^2$).

The following table shows how the 25 available marks are distributed:

Marks Awarded	Bounds on N	Additional Constraints
2 marks	$1 \leq N \leq 8$	None.
3 marks	$1 \leq N \leq 5\,000$	The original array A is a permutation.
5 marks	$1 \leq N \leq 3 \times 10^5$	The original array A is a permutation.
5 marks		None.
5 marks	$1 \leq N \leq 5 \times 10^6$	The original array A is a permutation.
5 marks		None.

Output Specification

Output N space-separated integers, the array $A[1], \dots, A[N]$, where $1 \leq A[i] \leq N$. It is guaranteed that there will always exist at least one valid array A .

If there is more than one valid array, you may output any valid array. In particular, even if the original array A is a permutation, your answer does not have to be a permutation.

Sample Input 1

3
3 1 2

Output for Sample Input 1

1 3 2

Explanation of Output for Sample Input 1

- The subarrays $[1, 3, 2]$, $[1, 3]$, $[1]$ have minimum 1. There are 3 such subarrays.
- The subarray $[3]$ has minimum 3. There is 1 such subarray.
- The subarrays $[3, 2]$ and $[2]$ have minimum 2. There are 2 such subarrays.

Sample Input 2

2
2 2

Output for Sample Input 2

1 1

Sample Input 3

3
1 4 1

Output for Sample Input 3

2 1 3

Explanation of Output for Sample Input 3

Note that $A = [2, 1, 2]$ would also be accepted by the judge.

2026 Canadian Computing Olympiad
Day 1, Problem 3
Beyond Counting

Time Limit: 5 seconds

Memory Limit: 1024 MB

Andy Jiang is studying data structures. One day, his friend Austin Zhu gave him a task on trees.

Austin provided a tree with N vertices, numbered from 1 to N . Each vertex i has a value A_i .

For each query, Austin asked Andy to consider a path between two vertices s_i and t_i , and compute how many times a given value x_i appears on that path.

Andy glanced at the problem and thought that this task was too easy for him.

Instead of just counting occurrences, Andy decided to challenge himself further. For each query, he wants to know how the frequency of x_i compares to other values on the same path.

Formally, for each query (s_i, t_i, x_i) :

- Consider the simple path from s_i to t_i .
- Let $\text{cnt}(y)$ be the number of occurrences of value y on this path.

Andy defines the *rank* of x_i as

$$1 + |\{y \mid \text{cnt}(y) > \text{cnt}(x_i)\}|.$$

That is, one plus the number of distinct values that appear more frequently than x_i on the path. Note that it is possible the value x_i does not appear on the path, i.e. $\text{cnt}(x_i) = 0$. In this case, you should return 1 plus the number of distinct values on the path.

In some test cases, the queries are given in an encoded form as described below.

Help Andy compute the rank of x_i for each query.

Input Specification

The first line contains three positive integers N , Q , and T ($1 \leq N, Q \leq 10^5$, $T \in \{0, 1\}$).

The second line contains N integers A_1, A_2, \dots, A_N ($1 \leq A_i \leq 10^9$).

The next $N - 1$ lines each contain two integers u_i, v_i ($1 \leq u_i, v_i \leq N$), representing the i -th edge.

Each of the next Q lines contains three integers $\hat{s}_i, \hat{t}_i, \hat{x}_i$ ($1 \leq \hat{s}_i, \hat{t}_i \leq N$, $1 \leq \hat{x}_i \leq 10^9$), describing the i -th query.

Let $\text{last}_0 = 0$. For each query $i = 1, 2, \dots, Q$, the actual parameters are defined as:

$$\begin{aligned} s_i &= ((\hat{s}_i + \text{last}_{i-1} \times T - 1) \bmod N) + 1, \\ t_i &= ((\hat{t}_i + \text{last}_{i-1} \times T - 1) \bmod N) + 1, \\ x_i &= ((\hat{x}_i + \text{last}_{i-1} \times T - 1) \bmod 10^9) + 1. \end{aligned}$$

After computing the answer to the i -th query, set

$$\text{last}_i = \text{answer to the } i\text{-th query.}$$

It may also be useful to note that “mod” corresponds to the % operator in most programming languages, indicating the remainder after division. For example, $5 \bmod 3 = 2$ and $17 \bmod 4 = 1$.

The following table shows how the 25 available marks are distributed:

Marks Awarded	Bounds on N, Q	Bounds on T	Additional Constraints
1 mark	$1 \leq N, Q \leq 10^3$	$T = 1$	None.
1 mark	$1 \leq N, Q \leq 10^5$	$T = 0$	All s_i are equal.
4 marks		$T = 1$	
4 marks		$T = 0$	$u_i = i$ and $v_i = i + 1$.
5 marks		$T = 1$	
3 marks		$T = 0$	
7 marks		$T = 1$	None.

Output Specification

For each query, output the answer to the query on a new line.

Sample Input 1

```
5 5 0
1 2 3 4 4
4 3
2 5
1 3
3 2
4 5 3
4 5 4
4 5 5
1 5 1
1 5 4
```

Output for Sample Input 1

```
2
1
4
1
1
```

Sample Input 2

```
5 5 1
1 2 3 4 4
4 3
2 5
1 3
3 2
4 5 3
2 3 2
3 4 4
2 1 999999997
5 4 3
```

Output for Sample Input 2

```
2
1
4
1
1
```

2026 Canadian Computing Olympiad
Day 2, Problem 1
Asymmetry

Time Limit: 2 seconds

Memory Limit: 512 MB

Alice and Bob will play a game with a grid with N rows and M columns where M is even. There is also a positive integer K . Initially, each cell of the grid contains a value between 0 and K , inclusive, and each cell is **unmarked**. The players take turns alternately, **with Alice going first**. The game ends when the current player cannot make a move.

On each player's turn, they select an **unmarked** cell of the grid and an integer a between 0 and K , inclusive. Then, they set the value of that cell to a and **mark** all cells in the same column as the one selected (including the selected cell).

The *asymmetry* of the grid is the sum of the absolute differences between the value of one cell and its corresponding cell when **horizontally** reflected across the middle of the grid over all such pairs of cells. More formally, it is

$$\sum_{1 \leq i \leq N} \left(\sum_{1 \leq j \leq M/2} |g_{i,j} - g_{i,M-j+1}| \right),$$

where $g_{i,j}$ is the value of the cell in the i -th row from the top and j -th column from the left. For example, the following 2×4 grid has an asymmetry of $|8-0| + |4-2| + |6-6| + |7-9| = 12$.

8	4	2	0
6	7	9	6

Alice wishes to minimize the asymmetry of the grid when the game ends while Bob wishes to maximize it. If both players play optimally, what is the asymmetry of the final grid?

Input Specification

The first line of input will consist of three space-separated integers N , M , and K ($1 \leq N, M \leq 2000$, M is even, $1 \leq K \leq 10^9$).

The next N lines each contain M integers, where the i -th line contains integers $g_{i,1} \dots, g_{i,M}$ ($0 \leq g_{i,j} \leq K$), the values of the cells from left to right in the i -th row from the top.

The following table shows how the 25 available marks are distributed:

Marks Awarded	Bounds on N	Bounds on M	Bounds on K
2 marks	$N = 1$	$2 \leq M \leq 2\,000$	$1 \leq K \leq 10^9$
3 marks	$1 \leq N \leq 2\,000$	$M = 2$	$K = 1$
3 marks			$1 \leq K \leq 10$
3 marks			$1 \leq K \leq 10^9$
4 marks		$2 \leq M \leq 2\,000$	$K = 1$
4 marks			$1 \leq K \leq 10$
6 marks			$1 \leq K \leq 10^9$

Output Specification

Output a single integer, the asymmetry of the final grid if Alice and Bob play optimally.

Sample Input 1

```
3 2 1
1 0
1 0
0 0
```

Output for Sample Input 1

```
2
```

Explanation of Output for Sample Input 1

There are only 2 columns, so each player will make 1 move. With Alice going first, she can perform the following moves:

- Select one of the cells with a value of 1 in the first column and set its value to 0. Then the optimal move for Bob is to set the value of the cell on the same row in the second column to 1. The final grid will be like the original grid with exactly one of the first two rows of 0s and 1s swapped. Such a grid has an asymmetry of $|1 - 0| + |0 - 1| + |0 - 0| = 2$.
- Select one of the cells in the second column and first two rows and set its value to 1. Then the optimal move for Bob is to set the value of the cell on the same row in the first column to 0. Again, the final grid will be like the original grid with exactly one of the first two rows of 0s and 1s swapped. Such a grid has an asymmetry of 2.
- Select one of the cells in the third row and set its value to 1. Then the optimal move for Bob can be to set the value of the other cell in the third row to 0. Note that the cell selected already had the value of 0, and that such a move is allowed. Then, the final grid will have a 0 and a 1 in each row, resulting in an asymmetry of 3.
- Select any cell and set its value to its current value. Then the optimal move for Bob is to set the value of the cell in the remaining unmarked column and third row to 1. The final grid will have a 0 and a 1 in each row, resulting in an asymmetry of 3.

We see that regardless of how Alice makes her move, Bob will be able to play in such a way that the asymmetry is at least 2. If Alice selects her first move optimally, she can ensure that Bob cannot make the final asymmetry more than 2. Thus, the asymmetry if both players play optimally is 2.

Sample Input 2

```
1 10 21
4 2 0 6 7 6 9 9 10 21
```

Output for Sample Input 2

```
55
```

Explanation of Output for Sample Input 2

There is only a single row, so for each move, the current player will select an unmarked cell and set it to any value between 0 and 21, inclusive. The game ends after each player has made 5 moves, resulting in all 10 cells being marked.

Sample Input 3

```
4 6 986754321
219759391 882760615 762656191 423465948 621463211 136889371
215621504 385106915 740086459 417915224 551800597 572994766
176308756 365311996 635683450 907755406 590000050 586083433
607011121 457147795 837558908 684766852 946836347 303039615
```

Output for Sample Input 3

```
3972378656
```

Explanation of Output for Sample Input 3

Note that the answer may not fit within a 32-bit integer.

2026 Canadian Computing Olympiad
Day 2, Problem 2
Tree Traversals

Time Limit: 4 seconds

Memory Limit: 512 MB

Yevin Kang has a tree with N vertices that are labelled with integers from 1 to N . A tree is an undirected connected graph that does not contain a cycle.

Let K be a positive integer. We define $f(K)$ as follows.

For any two vertices $1 \leq u, v \leq N$, let $d(u, v)$ denote the number of edges on the simple path connecting vertex u and vertex v . In particular, $d(u, u) = 0$ for all $1 \leq u \leq N$.

A permutation p_1, \dots, p_N of $1, \dots, N$ is *good* if all of the following conditions are satisfied.

- $d(p_{i-1}, p_i) \leq K$ for all $i = 2, \dots, N$.
- $d(1, p_i) \leq d(1, p_j)$ for all pairs of integers (i, j) with $1 \leq i < j \leq N$.

Then, $f(K)$ is the number of good permutations.

Yevin thinks this problem is too easy, so he gives you Q positive integers K_1, \dots, K_Q . He asks you to print the values of $f(K_1), f(K_2), \dots, f(K_Q)$, mod $10^9 + 7$.

It may also be useful to note that “mod” corresponds to the % operator in most programming languages, indicating the remainder after division. For example, $5 \bmod 3 = 2$ and $17 \bmod 4 = 1$.

Input Specification

Each test has multiple test cases.

The first line of the test contains one integer T ($1 \leq T \leq 5 \times 10^5$) — the number of test cases.

The first line of each test case contains two space-separated integers N, Q ($1 \leq Q \leq N \leq 5 \times 10^5$).

Each of the next $N - 1$ lines contains two space-separated integers u, v — indicating that there is an edge connecting u and v in the tree. It is guaranteed that the $N - 1$ edges form a tree.

The next line contains Q integers, K_1, \dots, K_Q — denoting the Q queries.

It is guaranteed that the sum of N over all test cases in a test (denoted by $\sum N$) does not exceed 5×10^5 .

The following table shows how the 25 available marks are distributed:

Marks Awarded	Bounds on $\sum N$	Bounds on Q	Bounds on K_i
2 marks	$1 \leq \sum N \leq 10$	$1 \leq Q \leq N$	$1 \leq K_i \leq N$
3 marks	$1 \leq \sum N \leq 5 \times 10^5$	$1 \leq Q \leq \min(2, N)$	$1 \leq K_i \leq \min(2, N)$
5 marks	$1 \leq \sum N \leq 3000$	$1 \leq Q \leq \min(5, N)$	$1 \leq K_i \leq N$
7 marks	$1 \leq \sum N \leq 5 \times 10^5$		
8 marks		$1 \leq Q \leq N$	

Output Specification

For each test case, output one line with Q space-separated integers — the values of $f(K_1), f(K_2), \dots, f(K_Q)$, mod $10^9 + 7$.

Sample Input

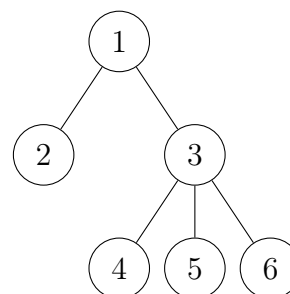
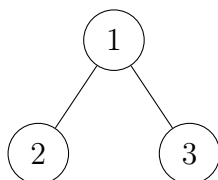
```
2
3 3
1 2
1 3
1 2 3
6 3
1 2
1 3
3 4
3 5
3 6
1 2 3
```

Output for Sample Input

```
0 2 2
0 6 12
```

Explanation of Output for Sample Input

The two trees in the sample input are shown below.



In the first test case, for $K = 2$ or $K = 3$, both $[1, 2, 3]$ and $[1, 3, 2]$ are good permutations. $[2, 1, 3]$ is not a good permutation for all values of K because

$$d(1, p_1) = 1 \not\leq 0 = d(1, p_2)$$

violates the second condition.

It can be shown that no permutation is good for $K = 1$.

In the second test case, $[1, 3, 2, 4, 5, 6]$ is a good permutation for $K = 3$ but not a good permutation for $K = 2$ because $d(2, 4) = 3 \not\leq 2$.

2026 Canadian Computing Olympiad
Day 2, Problem 3
Walking on a Graph

Time Limit: 5 seconds

Memory Limit: 512 MB

There is a graph with N nodes, numbered from 1 to N . Each node is coloured either black or white. Additionally, it is known that node 1 is black and node 2 is white. For any i and j where $i \neq j$, there exists a directed edge from node i to j that is either red or blue. Its colour is determined using the following logic:

- If $i < j$ and the nodes have the same colour, then it is red.
- If $i < j$ and the nodes have different colours, then it is blue.
- If $i > j$ and the nodes have the same colour, then it is blue.
- If $i > j$ and the nodes have different colours, then it is red.

LoBren's favourite colour is initially blue. He then takes a walk on the graph (note that walks allow for repeated vertices and edges). He uses the following rules when walking:

- If he is currently on node 1, his favourite colour becomes blue.
- Otherwise, if he is currently on node 2, his favourite colour becomes red.
- He then traverses an outgoing edge from his current node with the same colour as his favourite colour. It can be shown that such an edge must exist.
- Finally, he optionally repeats the process.

By writing down the nodes he visits, in order, he gets a list l_1, l_2, \dots, l_L . Find the number of possible lists, mod $10^9 + 7$, that satisfy the following conditions:

- The list starts at node 1 and ends at node 2.
- For all i where $3 \leq i \leq N$, node i appears at most once in the list.
- For all j where $3 \leq j \leq L$, we have $l_{j-2} \neq l_j$.

It is provable that the number of such lists is finite.

It may also be useful to note that “mod” corresponds to the % operator in most programming languages, indicating the remainder after division. For example, $5 \bmod 3 = 2$ and $17 \bmod 4 = 1$.

Input Specification

The first line of input contains a single integer, N .

The next line contains a string of length N , consisting of the characters B and W. If the i^{th} character is B, then node i is black. Otherwise, it is white. It is guaranteed that node 1 is black and node 2 is white.

The following table shows how the 25 available marks are distributed:

Marks Awarded	Bounds on N	Additional Constraints
1 mark	$3 \leq N \leq 8$	None.
3 marks	$3 \leq N \leq 20$	None.
4 marks	$3 \leq N \leq 50$	There exists exactly one black node.
4 marks		There exists an integer i where $2 \leq i \leq N$, such that every node in the range $[2, i]$ is white, and every other node is black.
6 marks		There exist at most 5 black nodes.
7 marks		None.

Output Specification

On a single line, output the number of possible lists, modulo $10^9 + 7$.

Sample Input 1

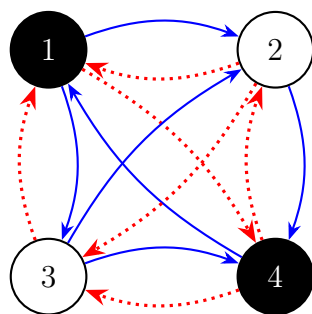
4
BWBB

Output for Sample Input 1

4

Explanation of Output for Sample Input 1

The graph looks like:



The solid lines represent blue edges, while the dashed lines represent red edges. The possible paths are:

- $1 \rightarrow \underline{2}$
- $1 \rightarrow 3 \rightarrow \underline{2}$
- $1 \rightarrow 3 \rightarrow 4 \rightarrow \underline{2}$
- $1 \rightarrow \underline{2} \dashrightarrow \underline{3} \dashrightarrow 1 \rightarrow \underline{2}$

The favourite colour is red at the underlined nodes, and blue otherwise.

Sample Input 2

12

BWBWBBBWBBW

Output for Sample Input 2

3377552