Problem A. Streets in Kaskelen

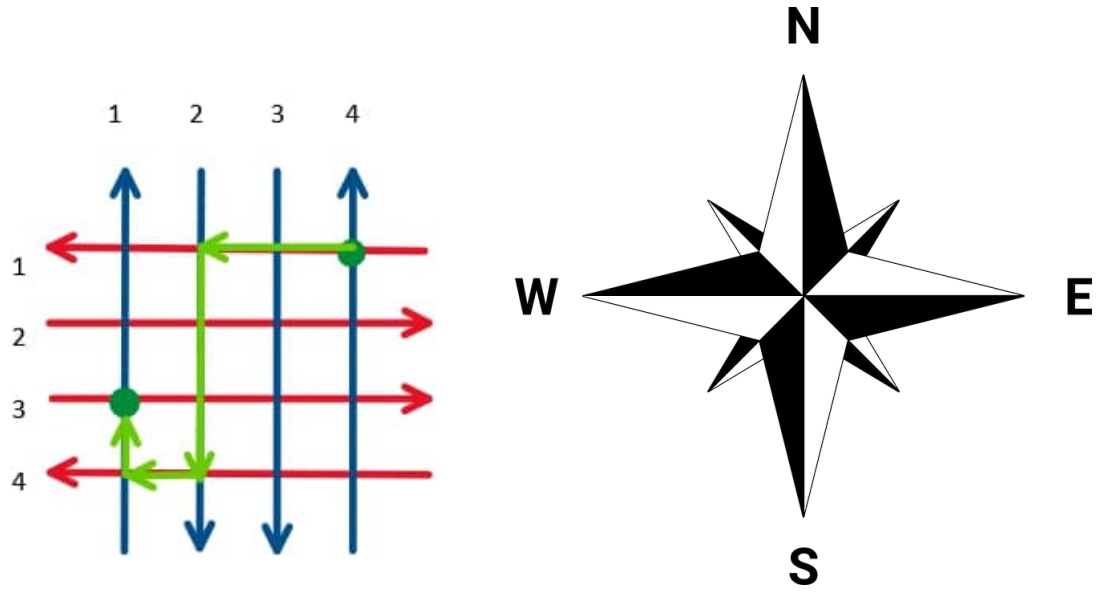
Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

Kaskelen city has n horizontal and n vertical streets which form a rectangular grid. Let's number from 1 to n the horizontal streets from north to south and the vertical streets from west to east. Denote the crossroad (i, j) as the crossroad on the intersection of i-th horizontal and j-th vertical streets.

Since the streets in Kaskelen are very busy, the city administration decided to make all streets one way. Such reform still have some downsides. Firstly, Kaskelen is a fast growing city and administration often has to change the direction of some streets. Secondly, sometimes it might happen, that it is impossible to reach some part of the city from another.

To keep track of such changes, you are asked to write a program that models this street system. You have to process queries of three types:

- 1 r_1 c_1 r_2 c_2 — check if it is possible to move from the crossroad (r_1, c_1) to crossroad (r_2, c_2).
- 2 r — The direction of the horizontal street r changes to the opposite.
- 3 c — The direction of the vertical street c changes to the opposite.



The image is matches the first example

Input

Each test consist of several test cases. The first line of the input contains one integer $t (1 \le t \le 1000)$ — the number of test cases. Then, descriptions of test cases follow.

The first line of a test case contains two integers n and $q (2 \le n \le 3 \cdot 10^5, 1 \le q \le 3 \cdot 10^5)$ — the number of horizontal/vertical streets and the number of queries.

The second line contains one string a of length n — the description of the horizontal streets. If a_i = 'L', then i-th road is directed from east to west. Otherwise if a_i = 'R', then from west to east.

The third line of a test case contains one string b of length n — description of vertical streets. If b_i = 'U', then i-th road is directed from south to north. Otherwise if b_i = 'D', then from north to south.

Each of the following q line contain a query, which is give in the format, described in the statement.

It is guaranteed, that the sum of values n over all test cases does not exceed $3 \cdot 10^5$.

It is also guaranteed, that the sum of values q over all test cases does not exceed $3 \cdot 10^5$.

Output

For each query of first type, print "YES" if it is possible to move from crossroad to another. Otherwise print "NO".

Scoring

Denote S as sum of n over all test cases. Denote T as sum of q over all test cases.

Subtask	Additional Constraints	Points	Necessary subtasks
0	Examples	0	—
1	$S \leq 10$, $T \leq 10^4$, no queries of second and third type	12	—
2	$S \leq 80$, $T \leq 2 \cdot 10^5$, no queries of second and third type	15	1
3	$a_1 = a_2 = \ldots = a_n$, no queries of second type	14	—
4	$S, T \leq 1000$, no queries of second and third type	16	—
5	$S, T \leq 50000$, no queries of second and third type	22	1, 4
6	—	21	0, 2, 3, 5

Example

standard input	standard output
1 4 4 LRRL UDDU 1 1 4 3 1 1 1 4 4 4 3 4 1 1 4 4 4	YES NO YES

Problem B. Challenges of urban planning

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

Citizens of Byteland have started building a new city Bittown by all of the modern standards of urban planning. Two famous urbanists — Adilkhan Paradoxny and Temirlan Bitihirovich were hired to design a new city's plan.

Bittown will have n crossroads and $n-1$ bidirectional streets connecting them. It is also guaranteed, that one can get from any crossroad to any other crossroad moving along the Bittown streets. Also, a single-family house will be built on each crossroad.

According to the plan, two schools will be built in Bittown. However, urbanists still need to choose two crossroads to build those schools. Notice that, if a school is built on a crossroad, a house with one family will still be built on that crossroad. It is also possible to build both schools on one crossroad.

Of course, it is important for urbanists, that new city locals will get to schools as quickly as possible. Each family will drive to school which is closest to their house.

Let's number crossroads from 1 to n and let $d(v,u)$ be the minimum number of streets required to get from crossroad v to crossroad u. Suppose, schools are built on crossroads with numbers a and b. Then, the level of inconvenience of schools $f(a,b)$ is defined as the sum of distances to the closest school over each household. Formally, $f(a,b) = \sum_{v=1}^{n} \min[d(a,v), d(b,v)]$.

Both urbanists are very proud and are not willing to discuss the design with each other. So, each of them will independently choose a future location of one of the schools.

Consider 3 possible scenarios:

1. You are responsible for choosing crossroads for both schools. In this case, find the smallest possible level of inconvenience $f(a,b)$ where $1 \le a, b \le n$.
2. Temirlan Bitihirovich wants to build a school on the crossroad $a = 1$, while Adilkhan Paradoxny asks you for help. Find the smallest possible level of inconvenience $f(a,b)$ when $1 \le b \le n$ and $a = 1$.
3. Adilkhan Paradoxny asks you for help, but Temirlan Bitihirovich did not disclose his plans. In this case, you need to find the smallest possible level of inconvenience $f(a,b)$, where $1 \le b \le n$ for each value of a from 1 to n.

Write a program that finds the smallest level of inconvenience in one of the described scenarios.

Input

The first line contains one integer t ($1 \le t \le 1000$) — number of test cases.

In following lines, descriptions of test cases are given.

First line of each test case contains two numbers n and p ($1 \le n \le 10^5$, $1 \le p \le 3$) — number of crossroads in Bittown and scenario number you are facing.

Following $n-1$ lines contain pairs (u_i, v_i) ($1 \le u_i, v_i \le n$, $u_i \ne v_i$, here $1 \le i \le n-1$) — indices of crossroads, connected by i-th road.

It is guaranteed, that the sum of values of n over all test cases does not exceed 10^5.

Output

For each test case given in input, in separate lines, print the answer in following format.

- If $p = 1$, print a single integer — the smallest possible value of $f(a, b)$.
- If $p = 2$, print a single integer — the smallest possible value of $f(a, b)$ when $a = 1$.
- If $p = 3$, print n integers $(e_1, \ldots, e_n)$, where e_i — the smallest possible value of $f(a, b)$ when $a = i$.

Scoring

Define S as the sum of n over all of test cases.

Subtask	Additional Constraints	Score	Necessary subtasks
0	Examples	0	—
1	$S \leq 500$	7	0
2	$(u_i, v_i) = (i, i + 1)$ for all $1 \leq i \leq n - 1$, $p = 3$	6	—
3	$S \leq 4000$	15	1
4	$p = 2$	11	—
5	$p = 1$	22	—
6	$S \leq 30000$	21	3
7	—	18	2, 4, 5, 6

Example

standard input	standard output
3 6 1 1 2 2 3 2 4 4 5 4 6 7 2 1 2 2 3 3 4 3 5 2 6 1 7 7 3 1 2 2 3 3 4 3 5 2 6 1 7	4 6 6 6 6 7 7 8 6

Note

In the first test case $p = 1$, the smallest value $f(a, b)$ is reached when $(a, b) = (2, 4)$. In this case, the level of inconvenience is equal to $1 + 0 + 1 + 0 + 1 + 1 = 4$.

In the second test case $p = 2$ and $a = 1$ is fixed, so the smallest value $f(a, b)$ is reached when $b = 3$. In this case, the level of inconvenience is equal to $0 + 1 + 0 + 1 + 1 + 2 + 1 = 7$.

Problem C. Green Line

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

There are n trees growing in one line on a largest street in Yesik city. Height of the i-th tree from the left is equal to a_i. At the end of each minute, the following transformation happens:

- If at least one neighbor of the tree i is taller, then the i-th tree will grow by 1 unit. More formally, at least one of the conditions $a_{i-1} > a_i$ or $a_{i+1} > a_i$ should be satisfied. Here we suppose that $a_0 = a_{n+1} = 0$.

Note that the transformation happens simultaneously for all trees. For example, if the current tree heights are equal to $[3, 3, 4, 2, 2]$, in the next minute heights will be equal to $[3, 4, 4, 3, 2]$.

The initial tree heights correspond to the 0-th minute. You have to answer q independent queries of the following type.

- What is the height of x-th tree at the beginning of t-th minute?

Input

First line contains two integers n and q ($2 \le n \le 10^5$, $1 \le q \le 10^5$) — number of trees and number of queries.

Second line contains n integers $a_1, \ldots, a_n$ ($1 \le a_i \le 10^{18}$ for all i, where $1 \le i \le n$) — initial heights of all trees.

Next q lines contain pairs (x_i, t_i) ($1 \le x_i \le n$, $0 \le t_i \le 10^{18}$) — descriptions of queries.

Output

Print q integers — answers to all queries in order or their appearance in the input.

Scoring

Subtask	Additional Constraints	Points	Necessary subtasks
0	Examples	0	—
1	$n \le 100$, $a_i \le 100$ for all $1 \le i \le n$	6	0
2	$n \le 100$	21	1
3	$n \le 2000$	20	2
4	—	53	3

Example

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

Note

Heights of all trees in the example are equal to:

- $[1, 3, 2, 5, 1]$ in the 0-th minute;
- $[2, 3, 3, 5, 2]$ in the 1-st minute;
- $[3, 3, 4, 5, 3]$ in the 2-nd minute;
- $[3, 4, 5, 5, 4]$ in the 3-rd minute