A - Learning Languages

CodeForces - 277A

The "BerCorp" company has got n employees. These employees can use m approved official languages for the formal correspondence. The languages are numbered with integers from 1 to m. For each employee we have the list of languages, which he knows. This list could be empty, i. e. an employee may know no official languages. But the employees are willing to learn any number of official languages, as long as the company pays their lessons. A study course in one language for one employee costs 1 berdollar.

Find the minimum sum of money the company needs to spend so as any employee could correspond to any other one (their correspondence can be indirect, i. e. other employees can help out translating).

Input

The first line contains two integers n and m ($2 \le n, m \le 100$) — the number of employees and the number of languages.

Then n lines follow — each employee's language list. At the beginning of the i-th line is integer k_i ($0 \le k_i \le m$) — the number of languages the i-th employee knows. Next, the i-th line contains k_i integers — a_{ij} ($1 \le a_{ij} \le m$) — the identifiers of languages the i-th employee knows. It is guaranteed that all the identifiers in one list are distinct. Note that an employee may know zero languages.

The numbers in the lines are separated by single spaces.

Output

Print a single integer — the minimum amount of money to pay so that in the end every employee could write a letter to every other one (other employees can help out translating).

Examples

Input

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

Output

```
0
```

Input

```
8 7
0
3 1 2 3
1 1
2 5 4
2 6 7
1 3
2 7 4
1 1
```

Output

```
2
```

Input

```
2 2
1 2
0
```

Output

```
1
```

Note

In the second sample the employee 1 can learn language 2, and emp
8 can learn language 4.

In the third sample employee 2 must learn language 2.

B - Marbles

Gym - 101908B

Using marbles as a currency didn't go so well in Cubicônia. In an attempt to make it up to his friends after stealing their marbles, the Emperor decided to invite them to a game night in his palace.

Of course, the game uses marbles, since the Emperor needs to find some use for so many of them. N marbles are scattered in a board whose lines are numbered from 0 through L and the columns numbered from 0 through C. Players alternate turns. In his turn, a player must choose one of the marbles and move it. The first player to move a marble to position $(0, 0)$ is the winner. The movements are limited so the game could be more interesting; otherwise, the first player could just move a marble to position $(0, 0)$ and win. A movement consists in choosing an integer u greater than 0 and a ball, whose location is denoted by (l, c), and move it to one of the following positions, as long as it is inside the board:

- $(l - u, c)$ or;
- $(l, c - u)$ or;
- $(l - u, c - u)$.

Note that more than one marble can occupy the same position on the board.

As the Emperor doesn't like to lose, you should help him determine which games he should attend. Also, as expected, the Emperor always take the first turn when playing. Assuming both players act optimally, you are given the initial distribution of the marbles, and should find if it is possible for the Emperor to win if he chooses to play.

Input

The first line contains an integer N ($1 \leq N \leq 1000$). Each of the following N rows contains two integers l_i and c_i indicating on whic row and column the i-th marble is in ($1 \leq l_i, c_i \leq 100$).

Output

Your program should print a single line containing the character Y if it is possible for the Emperor to win the game or N otherwise.

Examples

Input
2 1 3 2 3
Output
Y

Input
1 1 2
Output
N

C - Win or Freeze

CodeForces - 150A

You can't possibly imagine how cold our friends are this winter in Nvodsk! Two of them play the following game to warm up: initially a piece of paper has an integer q. During a move a player should write any integer number that is a non-trivial divisor of the last written number. Then he should run this number of circles around the hotel. Let us remind you that a number's divisor is called non-trivial if it is different from one and from the divided number itself.

The first person who can't make a move wins as he continues to lie in his warm bed under three blankets while the other one keeps running. Determine which player wins considering that both players play optimally. If the first player wins, print any winning first move.

Input

The first line contains the only integer q ($1 \le q \le 10^{13}$).

Please do not use the %lld specificator to read or write 64-bit integers in C++. It is preferred to use the cin, cout streams or the %I64d specificator.

Output

In the first line print the number of the winning player (1 or 2). If the first player wins then the second line should contain another integer — his first move (if the first player can't even make the first move, print 0). If there are multiple solutions, print any of them.

Examples

Input

6
Output
2

Input
30
Output
1 6

Input
1
Output
1 0

Note

Number 6 has only two non-trivial divisors: 2 and 3. It is impossible to make a move after the numbers 2 and 3 are written, so both of them are winning, thus, number 6 is the losing number. A player can make a move and write number 6 after number 30; 6, as we know, is a losing number. Thus, this move will bring us the victory.

D - Points

Gym - 102078B

You have been given the coordinates of N points in three-dimensional space—namely, $P_1, P_2, \ldots, P_N$. Initially, these points are distributed into N singleton sets, one for each point. You have been asked to execute Q operations involving these sets.

Operations may be of three distinct types:

- Type 1: given two points P_i and P_j, previously belonging to different sets, join the sets to which they belong.
- Type 2: undo the most recent union operation (Type 1) that has not already been undone. It is guaranteed that there exists at least one such operation.
- Type 3: given two points P_i and P_j, belonging to different sets, print the maximum Manhattan distance from a point in the same set as P_i to a point in the same set as P_j.

Input

The first line contains a single integer N, indicating the number of points which follow. Each of the next N lines contain three integers x_i, y_i and z_i, representing the coordinates of the ith point.

The next line contains an integer Q, indicating the number of queries that should be answered. Each of the next Q lines represents a query, and has one of the following three formats:

- $1\ i\ j$: represents a Type 1 query over the points P_i and P_j.
- 2: represents a Type 2 query.
- $3\ i\ j$: represents a Type 3 query over the points P_i and P_j.

Limites:

- $1 \leq N \leq 2 \times 10^5$
- $1 \leq Q \leq 3 \times 10^5$

- $-10^{\circ} \le x_i, y_i, z_i \le 10^{\circ}$

Output

For each Type 3 query, you should print its answer in a single line.

Examples

Input

```
5
1 5 0
2 4 0
3 3 0
4 2 0
5 1 0
7
3 1 2
1 1 4
3 1 2
1 2 5
3 1 2
2
3 1 2
```

Output

```
2
4
8
4
```

Input

```
4
-10 -10 0
-10 10 0
10 -10 0
10 10 0
6
3 2 4
1 1 2
3 2 4
```

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

Output

```
20
40
20
40
40
```

E - Mahmoud and a Dictionary

CodeForces - 766D

Mahmoud wants to write a new dictionary that contains n words and relations between them. There are two types of relations: synonymy (i. e. the two words mean the same) and antonymy (i. e. the two words mean the opposite). From time to time he discovers a new relation between two words.

He know that if two words have a relation between them, then each of them has relations with the words that has relations with the other. For example, if like means love and love is the opposite of hate, then like is also the opposite of hate. One more example: if love is the opposite of hate and hate is the opposite of like, then love means like, and so on.

Sometimes Mahmoud discovers a wrong relation. A wrong relation is a relation that makes two words equal and opposite at the same time. For example if he knows that love means like and like is the opposite of hate, and then he figures out that hate means like, the last relation is absolutely wrong because it makes hate and like opposite and have the same meaning at the same time.

After Mahmoud figured out many relations, he was worried that some of them were wrong so that they will make other relations also wrong, so he decided to tell every relation he figured out to his coder friend Ehab and for every relation he wanted to know is it correct or wrong, basing on the previously discovered relations. If it is wrong he ignores it, and doesn't check with following relations.

After adding all relations, Mahmoud asked Ehab about relations between some words based on the information he had given to him. Ehab is busy making a Codeforces round so he asked you for help.

\

Input

The first line of input contains three integers n, m and q ($2 \le n \le 10^5$, $m, q \le 10^5$) where n is the number of words in the dictionary, m is the number of relations Mahmoud figured out and q is the number of questions Mahmoud asked after telling all relations.

The second line contains n distinct words $a_1, a_2, \ldots, a_n$ consisting of small English letters with length not exceeding 20, which are the words in the dictionary.

Then m lines follow, each of them contains an integer t $(1 \le t \le 2)$ followed by two different words x_i and y_i which has appeared in the dictionary words. If $t = 1$, that means x_i has a synonymy relation with y_i, otherwise x_i has an antonymy relation with y_i.

Then q lines follow, each of them contains two different words which has appeared in the dictionary. That are the pairs of words Mahmoud wants to know the relation between basing on the relations he had discovered.

All words in input contain only lowercase English letters and their lengths don't exceed 20 characters. In all relations and in all questions the two words are different.

Output

First, print m lines, one per each relation. If some relation is wrong (makes two words opposite and have the same meaning at the same time) you should print "NO" (without quotes) and ignore it, otherwise print "YES" (without quotes).

After that print q lines, one per each question. If the two words have the same meaning, output 1. If they are opposites, output 2. If there is no relation between them, output 3.

See the samples for better understanding.

Examples

Input

```
3 3 4
hate love like
```

```
1 love like
2 love hate
1 hate like
love like
love hate
like hate
hate like
```

Output

```
YES
YES
NO
1
2
2
2
```

Input

```
8 6 5
hi welcome hello ihateyou goaway dog cat rat
1 hi welcome
1 ihateyou goaway
2 hello ihateyou
2 hi goaway
2 hi hello
1 hi hello
dog cat
dog hi
hi hello
ihateyou goaway
welcome ihateyou
```

Output

```
YES
YES
YES
YES
NO
YES
3
3
1
1
2
```

F - Strange Food Chain

SPOJ - CHAIN

There are 3 kinds of animals A,B and C. A can eat B,B can eat C,C can eat A. It's interesting,isn't it?

Now we have n animals,numbered from 1 to n. Each of them is one of the 3 kinds of animals:A,B,C.

Today Mary tells us k pieces of information about these n animals. Each piece has one of the two forms below:

- 1 x y: It tells us the kind of x and y are the same.
- 2 x y: It tells us x can eat y.

Some of these k pieces are true,some are false. The piece is false if it satisfies one of the 3 conditions below, otherwise it's true.

- X or Y in this piece is larger than n.
- This piece tells us X can eat X.
- This piece conflicts to some true piece before.

Input

The first line contains a single integer t.t blocks follow.

To every block,the first line contains two integers n(1<=n<=50000) and k (1<=k<=100000). k lines follow,each contains 3 positive integers D(1<=D<=2),X,Y,separated by single spaces.

Output

t lines,each contains a single integer - the number of false pieces in the corresponding block.

Example

Sample input:
```
1
100 7
1 101 1
2 1 2
2 2 3
2 3 3
1 1 3
2 3 1
1 5 5
```

Sample output:
```
3
```

Hint:

The false pieces are the 1st,the 4th and the 5th ones.

Warning: large Input/Output data, be careful with certain languages

G - Lieges of Legendre

CodeForces - 603C

Kevin and Nicky Sun have invented a new game called Lieges of Legendre. In this game, two players take turns modifying the game state with Kevin moving first. Initially, the game is set up so that there are n piles of cows, with the i-th pile containing a_i cows. During each player's turn, that player calls upon the power of Sunlight, and uses it to either:

1. Remove a single cow from a chosen non-empty pile.
2. Choose a pile of cows with even size $2 \cdot x$ ($x > 0$), and replace it with k piles of x cows each.

The player who removes the last cow wins. Given n, k, and a sequence a_1, a_2, ..., a_n, help Kevin and Nicky find the winner, given that both sides play in optimal way.

Input

The first line of the input contains two space-separated integers n and k ($1 \le n \le 100\,000, 1 \le k \le 10^9$).

The second line contains n integers, $a_1, a_2, \ldots a_n$ ($1 \le a_i \le 10^9$) describing the initial state of the game.

Output

Output the name of the winning player, either "Kevin" or "Nicky" (without quotes).

Examples

Input

```
2 1
3 4
```

Output
Kevin

Input
1 2 3
Output
Nicky

Note

In the second sample, Nicky can win in the following way: Kevin moves first and is forced to remove a cow, so the pile contains two cows after his move. Next, Nicky replaces this pile of size 2 with two piles of size 1. So the game state is now two piles of size 1. Kevin then removes one of the remaining cows and Nicky wins by removing the other.

H - Soteros

Gym - 101962J

Soteros is the biggest organized crime syndicate of Soteropolis. They are feared and respected. Members of Soteros follow a very rigid hierarchy. Sotero, the big boss, is the direct boss of some members, which in turn are direct boss of other members, and so on, like in a rooted tree structure. Therefore, Sotero is indirectly boss of the whole organization.

We can imagine a numbered rooted tree, where Sotero is represented by vertex 1. The other $n-1$ members are represented by numbers between 2 and n. The direct boss of the i-th member is p_i. Notice that every one has exactly one direct boss, except for Sotero.

Soteropolis is a city with K junctions and a bunch of two-way streets connecting pairs of these junctions. Soteros members have the culture of making themselves present on the streets. This is also true for the big boss Sotero. Every member of the organization is responsible for conducting business in at most one street of Soteropolis. More specifically, if the i-th member is responsible for some street, then junctions connected by this street are u_i, v_i.

Polo is an undercover agent working for SIA (Soteropolis Intelligence Agency) which finally got the chance to choose which member of Soteros he wants to work for. If he chooses to work for member i, then he will have free pass through all the streets that members (directly or indirectly) leadered by i are responsible for, but he won't be able to traverse any other streets of the town.

Polo definitely want to keep a low profile, but he won't be able to do much if he can't move around the city. He asked you to make an analysis to help on his decision.

A connected region of Soteropolis is a maximal set of junctions such that for every pair of these junctions there is a path between them consis only of streets Polo can freely traverse. In particular, an isolated junction (with no streets Polo can traverse around it) is a connected region.

For each member of Soteros, you should compute the number of connected regions Polo will be able to traverse if he chooses to work for such member.

Input

The first line contains two integers n, K $(2 \leq n, K \leq 10^5)$ – the number of members of Soteros and the number of junctions of Soteropolis, respectively.

The second line contains $n - 1$ integers separated by spaces. The i-th of them is p_{i+1} $(1 \leq p_{i+1} \leq i)$ – the direct boss of the $(i+1)$-th member.

The next n lines contains two integers each. If the i-th of these lines contains 0 0, then the i-th member is responsible for no street. Otherwise, it contains two integers u_i, v_i $(1 \leq u_i, v_i \leq K; u_i \neq v_i)$ – the streets which the i-th member is responsible for.

Two distinct members can be responsible for a street connecting the same pair of junctions u_i, v_i. That means they are responsible for the same street.

Output

Output n lines. The i-th of them should contain an integer – the number of connected regions Polo will be able to traverse if he chooses to work for member i.

Examples

Input
3 3 1 2 0 0 1 2 2 3

Output
1

```
1
2
```

Input

```
8 7
1 1 1 2 2 5 5
1 2
2 3
0 0
6 7
2 7
7 5
7 3
5 2
```

Output

```
2
4
7
```

I - Coins Game

SPOJ - MCOINS

Asen and Boyan are playing the following game. They choose two different positive integers K and L, and start the game with a tower of N coins. Asen always plays first, Boyan – second, after that – Asen again, then Boyan, and so on. The boy in turn can take 1, K or L coins from the tower. The winner is the boy, who takes the last coin (or coins). After a long, long playing, Asen realizes that there are cases in which he could win, no matter how Boyan plays. And in all other cases Boyan being careful can win, no matter how Asen plays.

So, before the start of the game Asen is eager to know what game case they have. Write a program coins which help Asen to predict the game result for given K, L and N.

INPUT

The input describes m games.

The first line of the standard input contains the integers K, L and m, $1 < K < L < 10$, $3 < m < 50$. The second line contains m integers $N1, N2, \ldots, Nm$, $1 \le Ni \le 1\,000\,000$, $i = 1, 2, \ldots, m$, representing the number of coins in each of the m towers

```
SAMPLE INPUT
2 3 5
3 12 113 25714 88888
```

OUTPUT

The standard output contains a string of length m composed of letters A and B. If Asen wins the ith game (no matter how the opponent plays), the ith letter of the string has to be A. When Boyan wins the ith game (no matter how Asen plays), the ith letter of the string has to be B.

```
SAMPLE OUTPUT
ABAAB
```

J - Galaksija

Kattis - galaksija

A long time ago in a galaxy far, far away, there were N planets. There were also $N-1$ interplanetary paths that connected all the planets (directly or indirectly). In other words, the network of planets and paths formed a tree. Additionally, each path was enumerated with an integer that denoted the curiosity of the path.

A pair of planets A, B is boring if the following holds:

- A and B are different planets;
- travelling between planet A and B is possible using one or more interplanetary paths; and
- the binary XOR of the curiosity of all the paths in that travel is equal to 0

Alas, the times have changed and an evil emperor is ruling the galaxy. He decided to use the Force to destroy all the interplanetary paths in a certain order. Determine the number of boring pairs of planets before the emperor started the destruction and after each destruction.

Input

The first line of input contains the integer N ($1 \leq N \leq 100\,000$). Each of the following $N-1$ lines contains three integers A_i, B_i, Z_i ($1 \leq A_i, B_i \leq N, 0 \leq Z_i \leq 1\,000\,000\,000$) that denote that planets A_i and B_i are directly connected with a path of curiosity Z_i. The following line of input contains the permutation of the first $N-1$ integers that denote the order in which the emperor is destroying the paths. If the i-th element of the permutation is j, then the emperor destroyed the path between planets A_j and B_j in the i-th step.

Output

The output must contain N lines, the k-th line containing the num
of boring pairs A, B from the task after the emperor destroyed exactly

$k - 1$ paths.

Sample Input 1

```
2
1 2 0
1
```

Sample Output 1

```
1
0
```

Sample Input 2

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

Sample Output 2

```
1
0
0
```

Sample Input 3

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

Sample Output 3

```
6
3
1
0
```

K - Pictionary

Gym - 102078A

There is a planet, in a yet undiscovered part of the universe, with a country inhabited solely by mathematicians. In this country, there are a total of N mathematicians, and the interesting fact is that each mathematician lives in their own city. Is it also interesting that no two cities are connected with a road, because mathematicians can communicate online or by reviewing academic papers. Naturally, the cities are labeled with numbers from 1 to N.

Life was perfect until one mathematician decided to write an academic paper on their smartphone. The smartphone auto-corrected the word "self-evident" to "Pictionary" and the paper was published as such. Soon after, the entire country discovered pictionary and wanted to meet up and play, so construction work on roads between cities began shortly.

The road construction will last a total of M days, according to the following schedule: on the first day, construction is done on roads between all pairs of cities that have M as their greatest common divisor. On the second day, construction is done on roads between all pairs of cities that have $M - 1$ as their greatest common divisor, and so on until the Mth day when construction is done on roads between all pairs of cities that are co-prime. More formally, on the ith day, construction is done on roads between cities a and b if $\gcd(a, b) = M - i + 1$.

Since the mathematicians are busy with construction work, they've asked you to help them determine the minimal number of days before a given pair of mathematicians can play pictionary together.

Input

The first line of input contains three positive integers N, M and Q ($1 \leq N, Q \leq 10^5, 1 \leq M \leq N$), the number of cities, the number (days it takes to build the roads, and the number of queries.

Each of the following Q lines contains two distinct positive integers A and B ($1 \leq A, B \leq N$) that denote the cities of the mathematicians who want to find out the minimal number of days before they can play pictionary together.

Output

The ith line must contain the minimal number of days before the mathematicians from the ith query can play pictionary together.

Examples

Input

```
8 3 3
2 5
3 6
4 8
```

Output

```
3
1
2
```

Input

```
25 6 1
20 9
```

Output

```
4
```

Input

```
9999 2222 2
1025 2405
3154 8949
```

Output

```
1980
2160
```

Note

On the first day, road (3, 6) is built. Therefore the answer to the second query is 1. On the second day, roads (2, 4), (2, 6), (2, 8), (4, 6) and (6, 8) are built. Cities 4 and 8 are now connected (it is possible to get from the first to the second using city 6). On the third day, roads between relatively prime cities are built, so cities 2 and 5 are connected.

On the second day, road (20, 15) is built, whereas on the fourth day, road (15, 9) is built. After the fourth day, cities 20 and 9 are connected via city 15.