CODEFORCES β
Sponsored by Telegram

Manthan, Codefest 17

A. Tom Riddle's Diary

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Harry Potter is on a mission to destroy You-Know-Who's Horcruxes. The first Horcrux that he encountered in the Chamber of Secrets is Tom Riddle's diary. The diary was with Ginny and it forced her to open the Chamber of Secrets. Harry wants to know the different people who had ever possessed the diary to make sure they are not under its influence.

He has names of n people who possessed the diary in order. You need to tell, for each person, if he/she possessed the diary at some point before or not.

Formally, for a name s_i in the i-th line, output "YES" (without quotes) if there exists an index j such that $s_i = s_j$ and $j < i$, otherwise, output "NO" (without quotes).

Input

First line of input contains an integer n $(1 \le n \le 100)$ — the number of names in the list.

Next n lines each contain a string s_i, consisting of lowercase English letters. The length of each string is between 1 and 100.

Output

Output n lines each containing either "YES" or "NO" (without quotes), depending on whether this string was already present in the stream or not.

You can print each letter in any case (upper or lower).

Examples

input
6 tom lucius ginny harry ginny harry

output
NO NO NO NO YES YES

input
3 a a a

output
NO YES YES

Note

In test case 1, for $i = 5$ there exists $j = 3$ such that $s_i = s_j$ and $j < i$, which means that answer for $i = 5$ is "YES".

B. Marvolo Gaunt's Ring

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Professor Dumbledore is helping Harry destroy the Horcruxes. He went to Gaunt Shack as he suspected a Horcrux to be present there. He saw Marvolo Gaunt's Ring and identified it as a Horcrux. Although he destroyed it, he is still affected by its curse. Professor Snape is helping Dumbledore remove the curse. For this, he wants to give Dumbledore exactly x drops of the potion he made.

Value of x is calculated as maximum of $p \cdot a_i + q \cdot a_j + r \cdot a_k$ for given p, q, r and array $a_1, a_2, \ldots a_n$ such that $1 \le i \le j \le k \le n$. Help Snape find the value of x. Do note that the value of x may be negative.

Input

First line of input contains 4 integers n, p, q, r ($-10^9 \le p, q, r \le 10^9$, $1 \le n \le 10^5$).

Next line of input contains n space separated integers $a_1, a_2, \ldots a_n$ ($-10^9 \le a_i \le 10^9$).

Output

Output a single integer the maximum value of $p \cdot a_i + q \cdot a_j + r \cdot a_k$ that can be obtained provided $1 \le i \le j \le k \le n$.

Examples

input
5 1 2 3 1 2 3 4 5
output
30

input
5 1 2 -3 -1 -2 -3 -4 -5
output
12

Note

In the first sample case, we can take $i = j = k = 5$, thus making the answer as $1 \cdot 5 + 2 \cdot 5 + 3 \cdot 5 = 30$.

In second sample case, selecting $i = j = 1$ and $k = 5$ gives the answer 12.

C. Helga Hufflepuff's Cup

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Harry, Ron and Hermione have figured out that Helga Hufflepuff's cup is a horcrux. Through her encounter with Bellatrix Lestrange, Hermione came to know that the cup is present in Bellatrix's family vault in Gringott's Wizarding Bank.

The Wizarding bank is in the form of a tree with total n vaults where each vault has some type, denoted by a number between 1 to m. A tree is an undirected connected graph with no cycles.

The vaults with the highest security are of type k, and all vaults of type k have the highest security.

There can be at most x vaults of highest security.

Also, **if a vault is of the highest security, its adjacent vaults are guaranteed to not be of the highest security and their type is guaranteed to be less than k**.

Harry wants to consider every possibility so that he can easily find the best path to reach Bellatrix's vault. So, you have to tell him, given the tree structure of Gringotts, the number of possible ways of giving each vault a type such that the above conditions hold.

Input

The first line of input contains two space separated integers, n and m — the number of vaults and the number of different vault types possible. ($1 \le n \le 10^5$, $1 \le m \le 10^9$).

Each of the next $n - 1$ lines contain two space separated integers u_i and v_i ($1 \le u_i, v_i \le n$) representing the i-th edge, which shows there is a path between the two vaults u_i and v_i. It is guaranteed that the given graph is a tree.

The last line of input contains two integers k and x ($1 \le k \le m$, $1 \le x \le 10$), the type of the highest security vault and the maximum possible number of vaults of highest security.

Output

Output a single integer, the number of ways of giving each vault a type following the conditions modulo $10^9 + 7$.

Examples

input
4 2 1 2 2 3 1 4 1 2
output
1

input
3 3 1 2 1 3 2 1
output
13

input
3 1 1 2 1 3 1 1
output
0

Note

In test case 1, we cannot have any vault of the highest security as its type is 1 implying that its adjacent vaults would have to have a vault type less than 1, which is not allowed. Thus, there is only one possible combination, in which all the vaults have type 2.

D. Rowena Ravenclaw's Diadem

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Harry, upon inquiring Helena Ravenclaw's ghost, came to know that she told Tom Riddle or You-know-who about Rowena Ravenclaw's diadem and that he stole it from her.

Harry thought that Riddle would have assumed that he was the only one to discover the Room of Requirement and thus, would have hidden it there. So Harry is trying to get inside the Room of Requirement to destroy the diadem as he knows that it is a horcrux.

But he has to answer a puzzle in order to enter the room. He is given n objects, numbered from 1 to n. Some of the objects have a parent object, that has a lesser number. Formally, object i may have a parent object $parent_i$ such that $parent_i < i$.

There is also a type associated with each parent relation, it can be either of type 1 or type 2. Type 1 relation means that the child object is like a special case of the parent object. Type 2 relation means that the second object is **always** a part of the first object and all its special cases.

Note that if an object b is a special case of object a, and c is a special case of object b, then c is considered to be a special case of object a as well. The same holds for parts: if object b is a part of a, and object c is a part of b, then we say that object c is a part of a. Also note, that if object b is a part of a, and object c is a special case of a, then b is a part of c as well.

An object is considered to be neither a part of itself nor a special case of itself.

Now, Harry has to answer two type of queries:

- `1 u v`: he needs to tell if object v is a special case of object u.
- `2 u v`: he needs to tell if object v is a part of object u.

Input

First line of input contains the number n ($1 \le n \le 10^5$), the number of objects.

Next n lines contain two integer $parent_i$ and $type_i$ ($-1 \le parent_i < i$ $parent_i \ne 0$, $-1 \le type_i \le 1$), implying that the i-th object has the parent $parent_i$. (If $type_i = 0$, this implies that the object i is a special case of object $parent_i$. If $type_i = 1$, this implies that the object i is a part of object $parent_i$). In case the i-th object has no parent, both $parent_i$ and $type_i$ are `-1`.

Next line contains an integer q ($1 \le q \le 10^5$), the number of queries.

Next q lines each represent a query having three space separated integers $type_i, u_i, v_i$ ($1 \le type_i \le 2$, $1 \le u, v \le n$).

Output

Output will contain q lines, each containing the answer for the corresponding query as "`YES`" (affirmative) or "`NO`" (without quotes).

You can output each letter in any case (upper or lower).

Examples

input
3 -1 -1 1 0 2 0 2 1 1 3 2 1 3
output
YES NO

input
3 -1 -1 1 0 1 1 2 2 2 3 2 3 2
output

```
YES
NO
```

Note

In test case 1, as object 2 is a special case of object 1 and object 3 is a special case of object 2, this makes object 3 a special case of object 1.

In test case 2, as object 2 is a special case of object 1 and object 1 has object 3, this will mean that object 2 will also have object 3. This is because when a general case (object 1) has object 3, its special case (object 2) will definitely have object 3.

E. Salazar Slytherin's Locket

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Harry came to know from Dumbledore that Salazar Slytherin's locket is a horcrux. This locket was present earlier at 12 Grimmauld Place, the home of Sirius Black's mother. It was stolen from there and is now present in the Ministry of Magic in the office of Dolorous Umbridge, Harry's former Defense Against the Dark Arts teacher.

Harry, Ron and Hermione are infiltrating the Ministry. Upon reaching Umbridge's office, they observed a code lock with a puzzle asking them to calculate count of magic numbers between two integers l and r (both inclusive).

Harry remembered from his detention time with Umbridge that she defined a magic number as a number which when converted to a given base b, all the digits from 0 to $b - 1$ appear even number of times in its representation without any leading zeros.

You have to answer q queries to unlock the office. Each query has three integers b_i, l_i and r_i, the base and the range for which you have to find the count of magic numbers.

Input

First line of input contains q ($1 \le q \le 10^5$) — number of queries.

Each of the next q lines contain three space separated integers b_i, l_i, r_i ($2 \le b_i \le 10$, $1 \le l_i \le r_i \le 10^{18}$).

Output

You have to output q lines, each containing a single integer, the answer to the corresponding query.

Examples

input
2 2 4 9 3 1 10
output
1 2

input
2 2 1 100 5 1 100
output
21 4

Note

In sample test case 1, for first query, when we convert numbers 4 to 9 into base 2, we get:

- $4 = 100_2$,
- $5 = 101_2$,
- $6 = 110_2$,
- $7 = 111_2$,
- $8 = 1000_2$,
- $9 = 1001_2$.

Out of these, only base 2 representation of 9 has even number of 1 and 0. Thus, the answer is 1.

F. Nagini

time limit per test: 4 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Nagini, being a horcrux You-know-who created with the murder of Bertha Jorkins, has accumulated its army of snakes and is launching an attack on Hogwarts school.

Hogwarts' entrance can be imagined as a straight line (x-axis) from 1 to 10^5. Nagini is launching various snakes at the Hogwarts entrance. Each snake lands parallel to the entrance, covering a segment at a distance k from $x = l$ to $x = r$. Formally, each snake can be imagined as being a line segment between points (l, k) and (r, k). Note that k can be both positive and negative, but not 0.

Let, at some x-coordinate $x = i$, there be snakes at point (i, y_1) and point (i, y_2), such that $y_1 > 0$ and $y_2 < 0$. Then, if for any point (i, y_3) containing a snake such that $y_3 > 0$, $y_1 \le y_3$ holds and for any point (i, y_4) containing a snake such that $y_4 < 0$, $|y_2| \le |y_4|$ holds, then the danger value at coordinate $x = i$ is $y_1 + |y_2|$. If no such y_1 and y_2 exist, danger value is 0.

Harry wants to calculate the danger value of various segments of the Hogwarts entrance. Danger value for a segment $[l, r)$ of the entrance can be calculated by taking the sum of danger values for each integer x-coordinate present in the segment.

Formally, you have to implement two types of queries:

- `1 l r k`: a snake is added parallel to entrance from $x = l$ to $x = r$ at y-coordinate $y = k$ (l inclusive, r exclusive).
- `2 l r`: you have to calculate the danger value of segment l to r (l inclusive, r exclusive).

Input

First line of input contains a single integer q ($1 \le q \le 5 \cdot 10^4$) denoting the number of queries.

Next q lines each describe a query. Each query description first contains the query type $type_i$ ($1 \le type_i \le 2$). This is followed by further description of the query. In case of the type being 1, it is followed by integers l_i, r_i and k_i ($1 \le l_i < r_i \le 10^5$, $-10^9 \le k_i \le 10^9$, $k \ne 0$). Otherwise, it just contains two integers, l_i and r_i ($1 \le l_i < r_i \le 10^5$).

Output

Output the answer for each query of type 2 in a separate line.

Examples

input
3 1 1 10 10 1 2 4 -7 2 1 10

output
34

input
7 1 2 3 5 1 1 10 10 1 4 5 -5 2 4 8 1 1 10 -10 2 4 8 2 1 10

output
15 75 170

Note

In the first sample case, the danger value for x-coordinates 1 is 0 as there is no y_2 satisfying the above condition for $x = 1$.

Danger values for x-coordinates 2 and 3 is $10 + |-7| = 17$.

Danger values for x-coordinates 4 to 9 is again 0 as there is no y_2 satisfying the above condition for these coordinates.

Thus, total danger value is $17 + 17 = 34$.

G. Harry Vs Voldemort

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

After destroying all of Voldemort's Horcruxes, Harry and Voldemort are up for the final battle. They each cast spells from their wands and the spells collide.

The battle scene is Hogwarts, which can be represented in the form of a tree. There are, in total, n places in Hogwarts joined using $n - 1$ undirected roads.

Ron, who was viewing this battle between Harry and Voldemort, wondered how many triplets of places (u, v, w) are there such that if Harry is standing at place u and Voldemort is standing at place v, their spells collide at a place w. This is possible for a triplet only when u, v and w are distinct, and there exist paths from u to w and from v to w which do not pass through the same roads.

Now, due to the battle havoc, new paths are being added all the time. You have to tell Ron the answer after each addition.

Formally, you are given a tree with n vertices and $n - 1$ edges. q new edges are being added between the nodes of the tree. After each addition you need to tell the number of triplets (u, v, w) such that u, v and w are distinct and there exist two paths, one between u and w, another between v and w such that these paths do not have an edge in common.

Input

First line contains an integer n ($1 \le n \le 10^5$), the number of places in Hogwarts.

Each of the next $n - 1$ lines contains two space separated integers u and v ($1 \le u, v \le n$) indicating a road between places u and v. It is guaranteed that the given roads form a connected tree.

Next line contains a single integer q ($1 \le q \le 10^5$), the number of new edges being added.

Each of the next q lines contains two space separated integers u and v ($1 \le u, v \le n$) representing the new road being added.

Note that it is possible that a newly added road connects places that were connected by a road before. Also, a newly added road may connect a place to itself.

Output

In the first line print the value for the number of triplets before any changes occurred.

After that print q lines, a single integer ans_i in each line containing the value for the number of triplets after i-th edge addition.

Examples

input
3 1 2 2 3 1 2 3
output
2 4

input
4 1 2 2 3 2 4 2 1 4 3 4
output
6 18 24

input
5 1 2

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

output

```
20
60
```

Note

In the first sample case, for the initial tree, we have $(1, 3, 2)$ and $(3, 1, 2)$ as the only possible triplets (u, v, w).

After addition of edge from 2 to 3, we have $(1, 3, 2)$, $(3, 1, 2)$, $(1, 2, 3)$ and $(2, 1, 3)$ as the possible triplets.