A. Sasha and the Beautiful Array

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

Sasha decided to give his girlfriend an array $a_1, a_2, \ldots, a_n$. He found out that his girlfriend evaluates the *beauty* of the array as the sum of the values $(a_i - a_{i-1})$ for all integers i from 2 to n.

Help Sasha and tell him the maximum beauty of the array a that he can obtain, if he can rearrange its elements in any way.

Input

Each test consists of multiple test cases. The first line contains a single integer t ($1 \le t \le 500$) — the number of test cases. The description of the test cases follows.

The first line of each test case contains a single integer n ($2 \le n \le 100$) — the length of the array a.

The second line of each test case contains n integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le 10^9$) — the elements of the array a.

Output

For each test case, output a single integer — the maximum beauty of the array a that can be obtained.

Standard Input	Standard Output
5 3 2 1 3 3 69 69 69 5 100 54 80 43 90 4 3 4 3 3 2 2 1	2 0 57 1 1

Note

In the first test case, the elements of the array a can be rearranged to make $a = [1, 2, 3]$. Then its beauty will be equal to $(a_2 - a_1) + (a_3 - a_2) = (2 - 1) + (3 - 2) = 2$.

In the second test case, there is no need to rearrange the elements of the array a. Then its beauty will be equal to 0.

B. Sasha and the Drawing

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

Even in kindergarten, Sasha liked a girl. Therefore, he wanted to give her a drawing and attract her attention.

As a drawing, he decided to draw a square grid of size $n \times n$, in which some cells are colored. But coloring the cells is difficult, so he wants to color as few cells as possible. But at the same time, he wants **at least** k diagonals to have at least one colored cell. Note that the square grid of size $n \times n$ has a total of $4n - 2$ diagonals.

Help little Sasha to make the girl fall in love with him and tell him the minimum number of cells he needs to color.

Input

Each test consists of multiple test cases. The first line contains a single integer t $(1 \leq t \leq 1000)$ — the number of test cases. The description of the test cases follows.

The only line of each test case contains two integers n and k $(2 \leq n \leq 10^8, 1 \leq k \leq 4n - 2)$ — the size of the square grid and the minimum number of diagonals in which there should be at least one colored cell.

Output

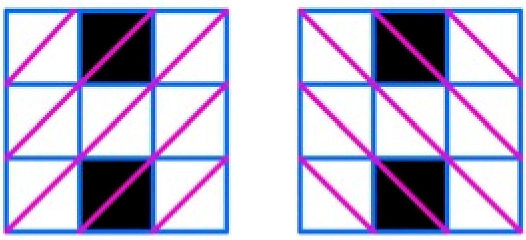
For each test case, output a single integer — the minimum number of cells that need to be colored.

Standard Input	Standard Output
7 3 4 3 3 3 10 3 9 4 7 7 11 2 3	2 2 6 5 4 6 2

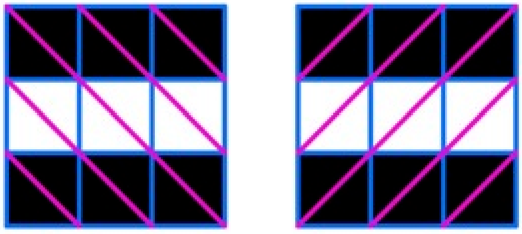
Note

In the pictures below, the colored cells are marked in black, and all diagonals are marked in purple.

In the first test case, you can color 2 cells so that 4 diagonals contain at least one colored cell:



In the third test case, you can color 6 cells so that all 10 diagonals contain at least one colored cell:



C. Sasha and the Casino

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

Sasha decided to give his girlfriend the best handbag, but unfortunately for Sasha, it is very expensive. Therefore, Sasha wants to earn it. After looking at earning tips on the internet, he decided to go to the casino.

Sasha knows that the casino operates under the following rules. If Sasha places a bet of y coins (where y is a positive integer), then in case of winning, he will receive $y \cdot k$ coins (i.e., his number of coins will increase by $y \cdot (k-1)$). And in case of losing, he will lose the entire bet amount (i.e., his number of coins will decrease by y).

Note that the bet amount must always be a positive (> 0) integer and cannot exceed Sasha's current number of coins.

Sasha also knows that there is a promotion at the casino: he cannot lose more than x times in a row.

Initially, Sasha has a coins. He wonders whether he can place bets such that he is guaranteed to win any number of coins. In other words, is it true that for any integer n, Sasha can make bets so that for any outcome that does not contradict the rules described above, at some moment of time he will have at least n coins.

Input

Each test consists of multiple test cases. The first line contains a single integer t ($1 \le t \le 1000$) — the number of test cases. The description of the test cases follows.

The single line of each test case contains three integers k, x and a ($2 \le k \le 30$, $1 \le x \le 100$, $1 \le a \le 10^9$) — the number of times the bet is increased in case of a win, the maximum number of consecutive losses, and the initial number of coins Sasha has.

Output

For each test case, output "YES" (without quotes) if Sasha can achieve it and "NO" (without quotes) otherwise.

You can output "YES" and "NO" in any case (for example, the strings "yEs", "yes" and "Yes" will be recognized as a positive answer).

Standard Input	Standard Output
9 2 1 7 2 1 1 2 3 15 3 3 6 4 4 5 5 4 7 4 88 1000000000 25 69 231 13 97 18806	YES NO YES NO NO YES NO NO NO

Note

In the first test case, Sasha can proceed as follows:

- If Sasha places a bet for the first time or if he won the previous bet, then he places 1 coin.
- If Sasha lost the previous bet, then he places 2 coins.

Note that Sasha cannot lose more than once in a row.

It can be proven that with this strategy, Sasha can obtain as many coins as he wants.

In the second test case, Sasha can only place 1 coin for the first time. But in case of a loss, he will not be able to place any more bets, so he will not be able to guarantee having as many coins as he wants.

,

D. Sasha and a Walk in the City

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

Sasha wants to take a walk with his girlfriend in the city. The city consists of n intersections, numbered from 1 to n. Some of them are connected by roads, and from any intersection, there is exactly one simple path[†] to any other intersection. In other words, the intersections and the roads between them form a tree.

Some of the intersections are considered dangerous. Since it is unsafe to walk alone in the city, Sasha does not want to visit three or more dangerous intersections during the walk.

Sasha calls a set of intersections *good* if the following condition is satisfied:

- If in the city only the intersections contained in this set are dangerous, then any simple path in the city contains **no more than two** dangerous intersections.

However, Sasha does not know which intersections are dangerous, so he is interested in the number of different good sets of intersections in the city. Since this number can be very large, output it modulo $998\,244\,353$.

[†] A simple path is a path that passes through each intersection at most once.

Input

Each test consists of multiple test cases. The first line contains a single integer t ($1 \le t \le 10^4$) — the number of test cases. The description of the test cases follows.

The first line of each test case contains a single integer n ($2 \le n \le 3 \cdot 10^5$) — the number of intersections in the city.

The next $(n - 1)$ lines describe the roads. The i-th line contains two integers u_i and v_i ($1 \le u_i, v_i \le n$, $u_i \ne v_i$) — the numbers of the intersections connected by the i-th road.

It is guaranteed that these roads form a tree.

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

Output

For each test case, output a single integer — the number of good sets of intersections modulo $998\,244\,353$.

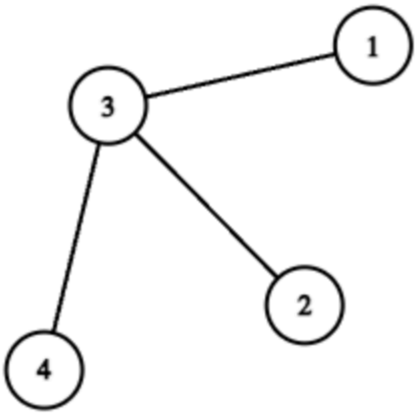
Standard Input	Standard Output
4 3 1 3 3 2 4 3 4 2 3 3 1 5	7 12 16 11

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

Note

In the first test case, there are $2^3 = 8$ sets of intersections. All of them are good, except for the set $\{1, 2, 3\}$, because if intersections $1, 2$, and 3 are dangerous, then the simple path $1 - 2 - 3$ contains 3 dangerous intersections. Thus, there are 7 good sets.

In the second test case, there are $2^4 = 16$ sets of intersections. Among them, the sets $\{1, 2, 3, 4\}$, $\{1, 2, 3\}$, $\{1, 3, 4\}$, $\{2, 3, 4\}$ are not good. Thus, there are a total of 12 good sets. The city layout is shown below:



E. Sasha and the Happy Tree Cutting

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

Sasha was given a tree† with n vertices as a prize for winning yet another competition. However, upon returning home after celebrating his victory, he noticed that some parts of the tree were missing. Sasha remembers that he colored some of the edges of this tree. He is certain that for any of the k pairs of vertices $(a_1, b_1), \ldots, (a_k, b_k)$, he colored at least one edge on the simple path‡ between vertices a_i and b_i.

Sasha does not remember how many edges he exactly colored, so he asks you to tell him the minimum number of edges he could have colored to satisfy the above condition.

† A tree is an undirected connected graph without cycles.

‡ A simple path is a path that passes through each vertex at most once.

Input

Each test consists of multiple test cases. The first line contains a single integer t ($1 \le t \le 10^4$) — the number of test cases. The description of the test cases follows.

The first line of each test case contains a single integer n ($2 \le n \le 10^5$) — the number of vertices in the tree.

The next $(n-1)$ lines describe the edges of the tree. The i-th line contains two integers u_i and v_i ($1 \le u_i, v_i \le n$, $u_i \ne v_i$) — the numbers of the vertices connected by the i-th edge.

The next line contains a single integer k ($1 \le k \le 20$) — the number of pairs of vertices between which Sasha colored at least one edge on a simple path.

The next k lines describe pairs. The j-th line contains two integers a_j and b_j ($1 \le a_j, b_j \le n, a_j \ne b_j$) — the vertices in the j-th pair.

It is guaranteed that the sum of n over all test cases does not exceed 10^5. It is guaranteed that the sum of 2^k over all test cases does not exceed 2^{20}.

Output

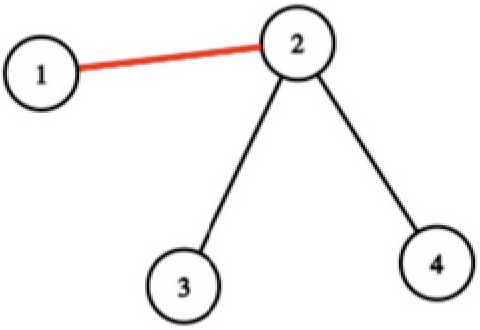
For each test case, output a single integer — the minimum number of edges Sasha could have colored.

Standard Input	Standard Output
3 4 1 2 2 3 2 4 2 1 3 4 1 6 1 2	1 2 4

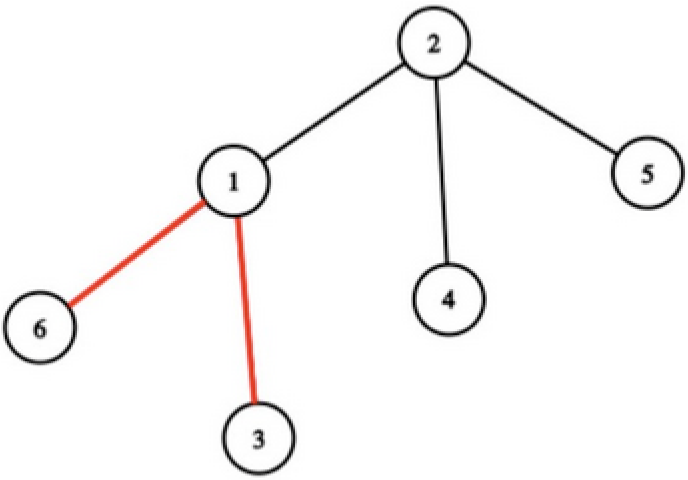
3 1 6 1 5 2 4 2 3 3 1 3 6 2 6 5 1 2 2 3 3 4 4 5 4 1 2 2 3 3 4 4 5	

Note

In the first test case, Sasha could have colored only one edge $(1, 2)$. Then, there would be at least one colored edge on the simple path between vertices 1 and 3, and vertices 4 and 1.



In the second test case, Sasha could have colored the edges $(1, 6)$ and $(1, 3)$.



F. Sasha and the Wedding Binary Search Tree

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

Having overcome all the difficulties and hardships, Sasha finally decided to marry his girlfriend. To do this, he needs to give her an engagement ring. However, his girlfriend does not like such romantic gestures, but she does like binary search trees†. So Sasha decided to give her such a tree.

After spending a lot of time on wedding websites for programmers, he found the perfect binary search tree with the root at vertex 1. In this tree, the value at vertex v is equal to val_v.

But after some time, he forgot the values in some vertices. Trying to remember the found tree, Sasha wondered — how many binary search trees could he have found on the website, if it is known that the values in all vertices are integers in the segment $[1, C]$. Since this number can be very large, output it modulo $998\,244\,353$.

†A binary search tree is a rooted binary tree in which for any vertex x, the following property holds: the values of all vertices in the left subtree of vertex x (if it exists) are less than or equal to the value at vertex x, and the values of all vertices in the right subtree of vertex x (if it exists) are greater than or equal to the value at vertex x.

Input

Each test consists of multiple test cases. The first line contains a single integer t ($1 \le t \le 10^5$) — the number of test cases. The description of the test cases follows.

The first line of each test case contains two integers n and C ($2 \le n \le 5 \cdot 10^5$, $1 \le C \le 10^9$) — the number of vertices in the tree and the maximum allowed value at the vertex.

The next n lines describe the vertices of the tree. The i-th line contains three integers L_i, R_i and val_i ($-1 \le L_i, R_i \le n$, $-1 \le val_i \le C$, $L_i, R_i, val_i \ne 0$) — the number of the left child, the number of the right child, and the value at the i-th vertex, respectively. If $L_i = -1$, then the i-th vertex has no left son. If $R_i = -1$, then the i-th vertex has no right son. If $val_i = -1$, then the value at the i-th vertex is unknown.

It is guaranteed that at least one suitable binary search tree exists.

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

Output

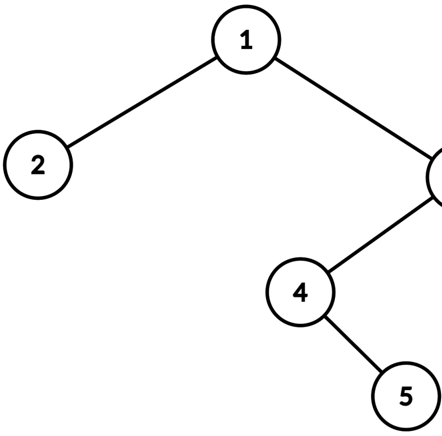
For each test case, output a single integer — the number of suitable binary search trees modulo $998\,244\,353$.

Standard Input	Standard Output
3 5 5 2 3 -1 -1 -1 2 4 -1 3 -1 5 -1 -1 -1 -1 3 69	4 1 10

2 3 47 -1 -1 13 -1 -1 69 3 3 2 3 -1 -1 -1 -1 -1 -1 -1	

Note

In the first test case, the binary search tree has the following form:



Then the possible values at the vertices are: $[2, 2, 3, 2, 2]$, $[2, 2, 3, 2, 3]$, $[2, 2, 3, 3, 3]$, and $[3, 2, 3, 3, 3]$.

In the second test case, the values at all vertices are known, so there is only one suitable binary search tree.