Coins on a Tree

Input file:	`standard input`
Output file:	`standard output`
Time limit:	8 seconds
Memory limit:	1024 megabytes

You are given a tree with n vertices, initially, there is exactly one coin in each vertex of the tree.

A total of $n-2$ operations will be performed. In each operation, one of the coins is removed from the tree.

After each operation, find the minimum distance between any two different coins on the tree and the number of pairs of coins that achieve this minimum distance. The distance between coins is defined as the number of edges on the simple path between the vertices containing these coins.

Input

The first line of input contains two integers n and t ($3 \le n \le 5 \cdot 10^5$, $0 \le t \le 1$) — the number of vertices in the tree, and a special parameter that determines how the queries are specified.

Each of the following $n-1$ lines of the input file contains two integers u and v, representing the next edge (u, v) in the tree ($1 \le u, v \le n$).

Each of the following $n-2$ lines contains an integer x'. The vertex number from which the coin is removed in this query is calculated using the formula: $x = x' \oplus (lastans \cdot t)$, where $\oplus$ denotes the bitwise XOR operation. Initially, $lastans = 0$, and after each query, $lastans$ equals the number of pairs from the answer to the previous query ($1 \le x \le n$, it is guaranteed that all values of x are distinct).

Output

Output $n-2$ lines, each containing two integers: the minimum distance and the number of pairs of vertices that achieve this minimum distance.

Scoring

Group	Points	Additional constraints			Required subgroups
		n	t	u and v	
1	11	$n \le 100$	$t = 0$	–	
2	13	$n \le 1000$	$t = 0$	–	1
3	28	$n \le 2 \cdot 10^5$	$t = 0$	–	1,2
4	15	$n \le 2 \cdot 10^5$	–	$u + 1 = v$	
5	16	$n \le 2 \cdot 10^5$	–	–	1-4
6	17	$n \le 5 \cdot 10^5$	–	–	1-5

Example

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