

Games

Input file: *standard input*
Output file: *standard output*
Time limit: 4 seconds
Memory limit: 1024 mebibytes

You are given a tree of n vertices. Since you feel that there is nothing to do, you want to play a game on this tree.

Before the game, you need to assign a label $\ell_u \in \{0, 1, 2, \dots, m-1, m\}$ to each vertex u .

The game consists of $m+1$ stages enumerated from 0 to m . In the i -th stage, all vertices u that satisfy $\ell_u \leq i$ will be painted black. If at this point, for every pair of **uncolored** vertices x and y , there exists a path from x to y that does not go through **any of the colored vertices**, then the game continues. Otherwise, you will lose and the game ends immediately. You win if the game continues after all stages.

You find that your ability to win the game depends only on how you initially assign labels to the vertices on this tree. In the next q days, you want to re-label the vertices and play the game. On the i -th day, you initially give the vertex a_i the label b_i . Then, you want to calculate how many ways are there to assign labels to the remaining vertices that allow you to win the game. Since the number could be large, you only need to output the answer modulo 998 244 353.

Input

The first line contains three integers n , m and q ($1 \leq n, q \leq 10^5$, $1 \leq m \leq 30$).

Each of the next $n-1$ lines contains two integers x and y ($1 \leq x, y \leq n$, $x \neq y$), indicating that there is an edge between vertices x and y . It is guaranteed that the given graph is a tree.

Each of the next q lines contains two integers a_i and b_i ($1 \leq a_i \leq n$, $0 \leq b_i \leq m$), indicating a query.

Output

For each query, output a single line containing a single integer, indicating the answer modulo 998 244 353.

Examples

<i>standard input</i>	<i>standard output</i>
3 2 5 1 2 3 2 3 1 1 0 2 0 2 1 2 2	7 9 5 8 9
6 6 6 6 1 1 3 2 6 4 2 5 2 2 1 3 2 5 3 5 6 5 3 1 1	896 2820 2433 1218 2433 2272