Problem O. Treeshop

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	512 megabytes

Max enjoys to play chess a lot. The only thing in this world that excites him more is to come up with ideas for programming contest's problems. This particular problem is of an ultimate delight for him as he was able to join his interests in one masterpiece.

Max has invented a tree-coordinate two-dimensional board and you are going to move a chip named treeshop along it. Formally, you are given two connected undirected acyclic graphs (also known as trees) of size n_1 and n_2 respectively. The position of a treeshop is defined by a pair of vertices (u, v), where u is some vertex of the first tree and v is some vertex of the second tree.

The distance between two vertices of the same tree equals to the number of edges the only simple path connecting these two vertices consists of. A treeshop is able to move from position (u, v) to position (u', v') in one move if and only if the distance between vertices u and u' equals to the distance between vertices v and v'. Obviously, if it is possible to get from (u, v) to (u', v') in one move, it is also possible to get from (u', v') to (u, v) in one move.

You are given q pairs of positions. For each of them you need to compute the minimum number of moves it will take a treeshop to get from one of these positions to the other.

Input

The first line of the input contains integers n_1, n_2 and q ($1 \le n_1, n_2, q \le 200\,000$) — the number of vertices in the first tree, the number of vertices in the second tree and the number of pairs of positions to process.

The following $n_1 - 1$ lines describe edges of the first tree. Each of them contains two integers u_i and v_i ($1 \le u_i, v_i \le n_1$, $u_i \ne v_i$) — the indices of vertices connected by the corresponding edge.

The following $n_2 - 1$ lines describe the second tree in exactly the same way.

Then follow q lines with the description of all pairs of positions to process. The i-th of these lines contains four integers s_1, s_2, t_1 and t_2 ($1 \le s_1, t_1 \le n_1$, $1 \le s_2, t_2 \le n_2$) — first two describe the initial position and the next two describe the target position.

Output

Print one integer for each query. If it is possible to get from the initial position to the target position by doing treeshop moves, print the minimum number of moves required to do so. Otherwise, print `-1`.

Example

standard input	standard output
4 5 7 1 2 2 3 2 4 1 2 2 3 3 4 4 5 1 1 2 5 1 5 4 1 1 5 4 2 2 5 2 3 2 1 2 5 3 2 3 2 4 4 1 2	-1 2 -1 2 3 0 1