

Perfect Journey

C/C++/Rust/Pascal 2024

C/C++/Rust/Pascal 512 M1024 M

64bit IO Format: %lld

🔗

In a country with  $n$  cities, there are  $n - 1$  bidirectional roads connecting these cities, forming a tree. You are now visiting this country, and there are  $m$  specific roads that you definitely want to travel through. The travel agency offers  $k$  optional tour routes. Each route starts from city  $s_i$  and follows the shortest path to reach city  $t_i$ .

Your goal is to select as few routes as possible from these  $k$  tour routes, ensuring that all  $m$  key roads are traveled at least once.

Please calculate the minimum number of tour routes you need to select, and the number of possible ways to achieve this minimum, with the answer modulo 998244353. A plan is defined as the set of tour routes you choose. Two plans are considered different if and only if there exists a tour route that is selected in one plan but not in the other.

If it's impossible to travel through all the specific roads, output  $-1$ .

The problem guarantees that the answer is non-zero modulo 998244353.

Input:

The first line contains three integers  $2 \leq n \leq 2 \times 10^5$ ,  $1 \leq m \leq 22$ ,  $1 \leq k \leq 2 \times 10^5$ , representing the number of cities, the number of specific roads, and the number of tour routes.

The next  $n - 1$  lines each contain two integers  $1 \leq u_i, v_i \leq n$ , guaranteeing that the given graph is a tree.

The next line contains  $m$  distinct integers  $1 \leq x_i \leq n - 1$  representing the indices of the specific roads (according to the input order).

The next  $k$  lines each contain two integers  $1 \leq s_i, t_i \leq n$ , indicating that the tour route goes from  $s_i$  to  $t_i$ .

Output:

Output the minimum number of tour routes you need to select, and the number of ways to achieve this minimum, with the answer modulo 998244353.

If it's impossible to travel through all the specific roads, output  $-1$ .

C++clang++18

1

ACM