

# Mountain Booking

Input file:            **standard input**  
Output file:           **standard output**  
Time limit:            6 seconds  
Memory limit:         512 megabytes

Byte Mountain is a beautiful tourist attraction in Byteland. There are  $n$  areas on the Byte Mountain, labeled by  $1, 2, \dots, n$ , connected by  $n - 1$  undirected roads. There is exactly one path between each pair of different areas. In other words, the map of the mountain is a tree with  $n$  nodes.

Assume today is day 0. In each of the next  $m$  days, the map of the mountain will be changed a little. Formally, in the early morning of day  $i$  ( $1 \leq i \leq m$ ), the  $k_i$ -th road will be closed forever, and another new road will be built, but the map of the mountain will always be a tree.

You are given the road construction plans for all the next  $m$  days, together with bookings of  $p$  tourists. The  $i$ -th tourist will visit the  $b_i$ -th area in the afternoon of day  $a_i$ , and will leave the mountain that night. You are now wondering about some interesting metrics.

Note that each road is weighted. Let  $f(u, v)$  be the maximum value of weights among all the roads on the unique single path from  $u$  to  $v$  ( $u \neq v$ ). You will be given  $q$  queries. In the  $i$ -th query, you will be given two integers  $c_i$  and  $d_i$ . You are required to calculate the following metric of the  $d_i$ -th area on the  $c_i$ -th day:

$$\sum_{1 \leq j \leq p, a_j = c_i, b_j \neq d_i} f(b_j, d_i)$$

## Input

The first line of the input contains four integers  $n, m, p$  and  $q$  ( $2 \leq n \leq 10^5, 1 \leq m, p, q \leq 2 \times 10^5$ ), denoting the number of areas, the number of incoming days, the number of tourists and the number of queries, respectively.

Each of the next  $n - 1$  lines contains three integers  $u_i, v_i$  and  $w_i$  ( $1 \leq u_i, v_i \leq n, u_i \neq v_i, 1 \leq w_i \leq 10^9$ ), denoting a two-way road between the  $u_i$ -th area and the  $v_i$ -th area, whose weight is  $w_i$ . The  $i$ -th road is labeled by  $i$  ( $1 \leq i < n$ ). These  $n - 1$  roads describe the map of the mountain at day 0.

Each of the next  $m$  lines contains four integers  $k_i, u_i, v_i$  and  $w_i$  ( $1 \leq k_i \leq n - 2 + i, 1 \leq u_i, v_i \leq n, u_i \neq v_i, 1 \leq w_i \leq 10^9$ ), denoting that at day  $i$  the road  $k_i$  will be closed forever, and the new road labeled by  $n - 1 + i$  will be built between the  $u_i$ -th area and the  $v_i$ -th area, whose weight is  $w_i$ . It is guaranteed that the map of the mountain will always be a tree, and no road will be closed more than once.

Each of the next  $p$  lines contains two integers  $a_i$  and  $b_i$  ( $1 \leq a_i \leq m, 1 \leq b_i \leq n$ ), describing a tourist's booking.

Each of the next  $q$  lines contains two integers  $c_i$  and  $d_i$  ( $1 \leq c_i \leq m, 1 \leq d_i \leq n$ ), describing a query.

## Output

For each query, print a single line containing an integer, denoting the answer.

## Example

standard input	standard output
5 3 6 6	8
1 2 9	3
1 3 4	9
1 4 6	11
4 5 2	8
3 3 5 3	0
2 1 5 5	
5 3 4 8	
1 3	
2 1	
3 3	
2 4	
1 5	
2 4	
1 1	
1 3	
2 5	
2 3	
3 1	
3 3	