
Problem A. Defense Tower

Input file: *standard input*
Output file: *standard output*
Time limit: 12 seconds
Memory limit: 512 mebibytes

In ICPCCamp, there are n cities conveniently labeled with $1, 2, \dots, n$, connected by $(n - 1)$ bidirectional roads. It is guaranteed that there is exactly one path between any two different cities.

In each city i , there is a defense tower with power a_i , built in the order $n, (n - 1), \dots, 1$. The towers are numbered the same as the cities. Therefore, tower n is the oldest tower while tower 1 is the newest. The *effect* of tower i on city j is defined as $eff(i, j) = a_i - \delta(i, j)$. Here, $\delta(i, j)$ is the number of roads between cities i and j . The *protector* of city j is the tower with maximum effect on it. If several towers have the same effect on a single city, the oldest one is chosen as the protector of this city.

Yuuka issues q commands to upgrade the power of the defense towers, where the k -th command is to add d_k points of power to the tower w_k . After each command, she would like to know the sum of protectors' labels for all cities. Note that the newly upgraded tower becomes the newest tower automatically.

However, there is a twist. Upgrading a tower is a costly operation. If the tower being upgraded is not even the protector for its own city, or $d_k = 0$, the upgrade command is ignored.

Input

The input contains zero or more test cases, and is terminated by end-of-file. For each test case:

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

The second line contains n integers a_1, a_2, \dots, a_n ($0 \leq a_i \leq 10^9$).

The i -th of the following $(n - 1)$ lines contains two integers u_i and v_i which denote a road between cities u_i and v_i ($1 \leq u_i, v_i \leq n$). It is guaranteed that there is exactly one path between any two different cities.

The k -th of the last q lines contains two integers w_k and d_k ($1 \leq w_k \leq n$, $0 \leq d_k \leq 10^9$).

It is guaranteed that both the sum of all n and the sum of all q do not exceed 10^5 .

Output

For each test case, output q integers s_1, s_2, \dots, s_q , where s_k denotes the sum of protectors' labels after the k -th command.

Example

standard input	standard output
3 3	4
1 1 0	4
1 3	4
2 3	8
1 2	8
2 2	
3 1000000000	
4 2	
2 4 4 4	
4 1	
4 2	
3 1	
2 4	
2 3	