Problem Description

In the mystical forest of Arboria, there stands an ancient tree of shrines. Each shrine i harbors a latent energy, denoted by a_i. Pilgrims begin their journey at a chosen shrine s. At each shrine they visit, if the energy there exceeds a certain tolerance T, the pilgrim becomes overwhelmed and ends their journey. Otherwise, the pilgrim proceeds to one of the direct child-shrines chosen uniformly at random; if there are no child-shrines, the journey ends naturally.

The tree's energy is not static – from time to time, rituals alter the energy of a single shrine. Your task is to assist the pilgrims by answering their queries: given a starting shrine s and a tolerance T, what is the expected number of shrines they will visit before their journey ends?

Formally, you are given a rooted tree with n nodes (the root is node 1). Each node i has a value a_i. Consider a random walk starting at node s: at each step, if the current node's value is greater than a given threshold T, the walk stops. Otherwise, the walk moves to a uniformly random child (if there are no children, it stops). You need to process two types of operations:

1. Update: change the value of a node x to v.
2. Query: given s and T, output the expected number of steps until the walk stops.

Input

The first line contains an integer n ($1 \leq n \leq 200,000$).
The second line contains n integers $a_1, a_2, \ldots, a_n$ ($1 \leq a_i \leq 10^9$).
The next $n-1$ lines each contain two integers u and v ($1 \leq u, v \leq n$), meaning v is a child of u. The edges form a tree rooted at 1.
The next line contains an integer q ($1 \leq q \leq 200,000$).
Then q lines follow, each describing an operation:

- `1 x v` ($1 \leq x \leq n$, $1 \leq v \leq 10^9$): update a_x to v.
- `2 s T` ($1 \leq s \leq n$, $1 \leq T \leq 10^9$): query the expected steps starting from s with threshold T.

Output

For each query of type 2, output the expected number of steps. Your answer is considered correct if the absolute or relative error is less than 10^{-6}.

Constraints

- $n \leq 200,000$
- $q \leq 200,000$
- $1 \leq a_i, v, T \leq 10^9$
- Time limit: 2 seconds
- Memory limit: 256 MB

Sample

Input:

```
3
1 2 3
1 2
2 3
5
2 1 5
1 3 10
2 1 5
1 2 10
2 1 5
```

Output:

```
2.0000000000
2.0000000000
1.0000000000
```

Explanation: The tree is a chain $1 \to 2 \to 3$. Initially, $a = [1, 2, 3]$.

- First query: $s = 1, T = 5$. The walk always goes $1 \to 2 \to 3$ and stops at 3 (no children). Hence, exactly 2 steps are taken.
- Update: set $a_3 = 10$.
- Second query: $s = 1, T = 5$. The walk still goes $1 \to 2 \to 3$, but at node 3 the energy $10 > 5$ causes an immediate stop. The number of steps remains 2.
- Update: set $a_2 = 10$.
- Third query: $s = 1, T = 5$. Now at node 2 the energy $10 > 5$, so the walk stops after the first step from 1 to 2. Thus, the expected number of steps is 1.