OTHS Coding Competition 3 (Mock CCC) S5 - World Tree

Time limit: 4.0s **Memory limit:** 512M

Nahida is running a simulation of Irminsul, the world tree, which stores all information about the world. In this simulation, Irminsul consists of N nodes, where node i is directly below node p_i and node 1 is at the very top of the tree, and $N-1$ connections connecting the nodes. The connections are formed in a way such that below each node, there are exactly 0 or 2 nodes and it is also guaranteed that you can reach any node from any other node via the connections. Node i has a maximum capacity of A_i units of information.

You have been tasked with helping Nahida perform Q operations on Irminsul, in order. The operations are of the following types t_i:

1. Add v_i units of information to node x_i. If a node's maximum information capacity is ever exceeded, the extra information will be distributed evenly to the nodes below it. If no other nodes are below the current one, the extra information is deleted.
2. Query the amount of information in node x_i.

Constraints

$3 \le N, Q \le 5 \times 10^5$

$1 \le p_i, x_i \le N$

$1 \le A_i, v_i \le 10^9$

$t_i \in \{1, 2\}$

There are exactly 0 or 2 nodes directly below each node.

Subtask 1 [3/15]

$3 \le N, Q \le 1000$

Subtask 2 [6/15]

$p_i = \lfloor i/2 \rfloor$

Subtask 3 [6/15]

No additional constraints.

Input Specification

The first line of input contains 2 space separated integers, N and Q, the number of nodes and the number of operations respectively.

The second line of input contains $N-1$ space separated integers, p_i for $i = 2, 3, \ldots, N$, the node directly above node i.

The third line of input contains N space separated integers, A_i, the maximum information capacity of node i.

The next Q lines of input contains t_i, x_i, v_i for type 1 operations and t_i, x_i for type 2 operations.

Output Specification

For each type 2 operation, output one decimal on its own line, the answer to that query. Your answer will be considered correct if it has an absolute or relative error of less than 10^{-5}.

Sample Input 1

```
5 5
1 1 2 2
1 2 3 4 5
1 1 6
2 3
2 4
1 2 1
2 5
```

Sample Output 1

```
2.5
0.25
0.75
```

Explanation for Sample Output 1

After the first type 1 operation, node 1 is full with 1 unit of information. The extra 5 units of information are distributed evenly between node 2 and node 3. Node 2 then becomes full with 2 units of information and the extra 0.5 units of information is distributed evenly between node 4 and node 5.

After the second type 1 operation, the 1 unit of information added to node 2 gets distributed evenly between node 4 and node 5, both of which now have 0.75 units of information.

Sample Input 2

```
5 6
3 1 1 3
1 1 1 1 1
1 1 999
2 1
2 2
2 3
2 4
2 5
```

Sample Output 2

```
1
1
1
1
1
```