2445. Number of Nodes With Value One Premium

Medium | Topics | Companies | Hint

There is an **undirected** connected tree with `n` nodes labeled from `1` to `n` and `n - 1` edges. You are given the integer `n`. The parent node of a node with a label `v` is the node with the label `floor (v / 2)`. The root of the tree is the node with the label `1`.

- For example, if `n = 7`, then the node with the label `3` has the node with the label `floor(3 / 2) = 1` as its parent, and the node with the label `7` has the node with the label `floor(7 / 2) = 3` as its parent.

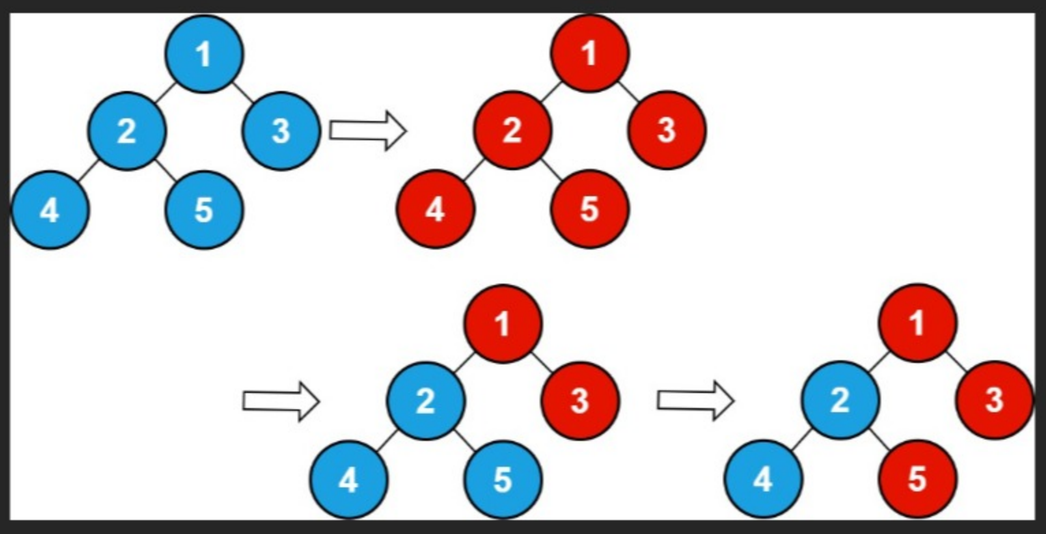
You are also given an integer array `queries`. Initially, every node has a value `0` on it. For each query `queries[i]`, you should flip all values in the subtree of the node with the label `queries[i]`.

Return *the total number of nodes with the value* `1` ***after processing all the queries***.

Note that:

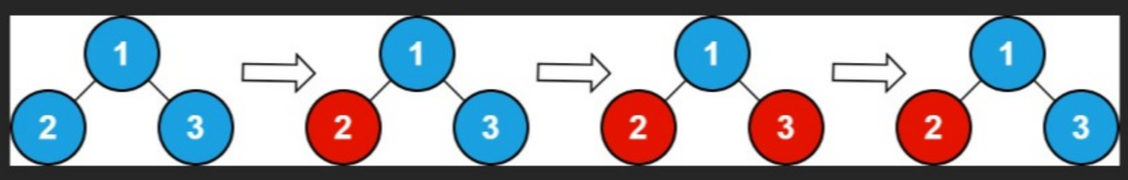
- Flipping the value of a node means that the node with the value `0` becomes `1` and vice versa.
- `floor(x)` is equivalent to rounding `x` down to the nearest integer.

Example 1:




```
Input: n = 5 , queries = [1,2,5]
Output: 3
Explanation: The diagram above shows the tree structure and its status after performing the queries. The blue node represents the value 0, and the red node represents the value 1.
After processing the queries, there are three red nodes (nodes with value 1): 1, 3, and 5.
```

Example 2:




```
Input: n = 3, queries = [2,3,3]
Output: 1
Explanation: The diagram above shows the tree structure and its status after performing the queries. The blue node represents the value 0, and the red node represents the value 1.
After processing the queries, there are one red node (node with value 1): 2.
```

Constraints:

- `1 <= n <= 10^5`
- `1 <= queries.length <= 10^5`
- `1 <= queries[i] <= n`

Seen this question in a real interview before? 1/5

Yes No

Accepted 2.2K | Submissions 3.2K | Acceptance Rate 68.1%

Topics

Tree Depth-First Search Breadth-First Search Binary Tree

Companies

0 - 6 months

Infosys 2

Hint 1

The relative order of processing queries does not matter.

Hint 2

If we know that for some node v, its parent was updated some number of times, then we know that node v was also updated that number of times.

Hint 3

Iterate on nodes from highest to lowest and count the number of times the query was performed on that node and the number of times this node was updated from its parent (direct or indirect). The parity of that number is the answer.

Discussion (1)