1938. Maximum Genetic Difference Query

My Submissions (/contest/weekly-contest-250/problems/maximum-genetic-difference-query/submissions/)

Back to Contest (/contest/weekly-contest-250/)

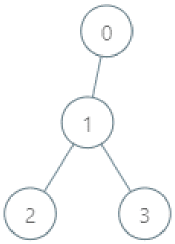
User Accepted:	190
User Tried:	897
Total Accepted:	210
Total Submissions:	1546
Difficulty:	Hard

There is a rooted tree consisting of `n` nodes numbered `0` to `n - 1`. Each node's number denotes its **unique genetic value** (i.e. the genetic value of node `x` is `x`). The **genetic difference** between two genetic values is defined as the **bitwise-XOR** of their values. You are given the integer array `parents`, where `parents[i]` is the parent for node `i`. If node `x` is the **root** of the tree, then `parents[x] == -1`.

You are also given the array `queries` where `queries[i]` = [node_i, val_i]. For each query `i`, find the **maximum genetic difference** between val_i and p_i, where p_i is the genetic value of any node that is on the path between node_i and the root (including node_i and the root). More formally, you want to maximize val_i XOR p_i.

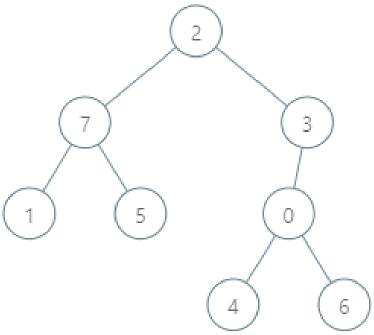
Return *an array* `ans` *where* `ans[i]` *is the answer to the* i^{th} *query*.

Example 1:



```
Input: parents = [-1,0,1,1], queries = [[0,2],[3,2],[2,5]]
Output: [2,3,7]
Explanation: The queries are processed as follows:
- [0,2]: The node with the maximum genetic difference is 0, with a difference of 2 XOR 0 = 2.
- [3,2]: The node with the maximum genetic difference is 1, with a difference of 2 XOR 1 = 3.
- [2,5]: The node with the maximum genetic difference is 2, with a difference of 5 XOR 2 = 7.
```

Example 2:



```
Input: parents = [3,7,-1,2,0,7,0,2], queries = [[4,6],[1,15],[0,5]]
Output: [6,14,7]
Explanation: The queries are processed as follows:
- [4,6]: The node with the maximum genetic difference is 0, with a difference of 6 XOR 0 = 6.
- [1,15]: The node with the maximum genetic difference is 1, with a difference of 15 XOR 1 = 14.
- [0,5]: The node with the maximum genetic difference is 2, with a difference of 5 XOR 2 = 7.
```

Constraints:

- 2 <= parents.length <= 10^5
- 0 <= parents[i] <= parents.length - 1 for every node i that is **not** the root.
- parents[root] == -1
- 1 <= queries.length <= 3 * 10^4
- 0 <= node$_i$ <= parents.length - 1
- 0 <= val$_i$ <= 2 * 10^5

Discuss (https://leetcode.com/problems/maximum-genetic-difference-query/discuss)

JavaScript

```
function Query(index, val) {
    this.index = index;
    this.val = val;
}

const maxGeneticDifference = (parents, queries) => {
    let pn = parents.length, qn = queries.length;
    let root = parents.indexOf(-1);
    let children = initializeGraph(pn);
    for (let i = 0; i < pn; i++) {
        if (i != root) {
            children[parents[i]].push(i);
        }
    }
    let freq = Array(1 << 20).fill(0);
    let queriesByNode = initializeGraph(pn);
    for (let i = 0; i < qn; i++) {
        let query = queries[i];
        queriesByNode[query[0]].push(new Query(i, query[1]));
    }
    let res = Array(qn).fill(0);
    const dfs = (idx) => {
        let y = (1 << 19) + idx;
        while (y > 0) {
            freq[y]++;
            y >>= 1;
        }
        for (const qnode of queriesByNode[idx]) {
            let j = qnode.index, x = qnode.val;
            let cum = 0;
            let bit = 1 << 18;
            while (bit > 0) {
                let ii = ((1 << 19) ^ cum ^ x ^ bit) / bit >> 0;
                if (freq[ii] > 0) cum += bit;
                bit >>= 1;
            }
            res[j] = cum;
        }
        for (const child of children[idx]) dfs(child);
        y = (1 << 19) + idx;
        while (y > 0) {
            freq[y]--;
            y >>= 1;
        }
    };
    dfs(root);
    return res;
};

const initializeGraph = (n) => {
    let G = [];
    for (let i = 0; i < n; i++) G.push([]);
    return G;
};
```

□ **Custom Testcase** Use Example Testcases

Run Submit

Submission Result: Accepted (/submissions/detail/524299495/) More Details (/submissions/detail/524299495/)

Share your acceptance!

Copyright © 2021 LeetCode

Help Center (/support) | Jobs (/jobs) | Bug Bounty (/bugbounty) | Online Interview (/interview/) | Students (/student) | Terms (/terms) | Privacy Policy (/privacy)

United States (/region)