8051. Maximum Number of K-Divisible Components

My Submissions (/contest/biweekly-contest-114/problems/maximum-number-of-k-divisible-components/submissions/)

Back to Contest (/contest/biweekly-contest-114/)

There is an undirected tree with `n` nodes labeled from `0` to `n - 1`. You are given the integer `n` and a 2D integer array `edges` of length `n - 1`, where `edges[i] = [a_i, b_i]` indicates that there is an edge between nodes `a_i` and `b_i` in the tree.

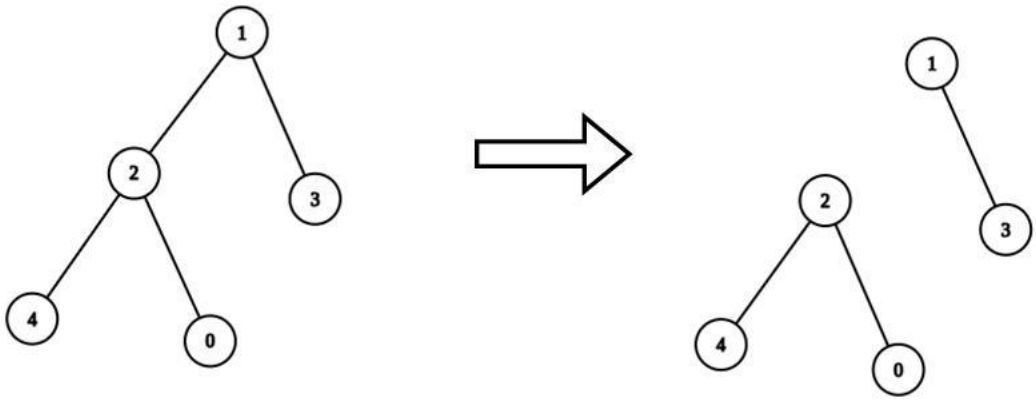
You are also given a **0-indexed** integer array `values` of length `n`, where `values[i]` is the **value** associated with the `i^th` node, and an integer `k`.

A **valid split** of the tree is obtained by removing any set of edges, possibly empty, from the tree such that the resulting components all have values that are divisible by `k`, where the **value of a connected component** is the sum of the values of its nodes.

Return *the **maximum number of components** in any valid split*.

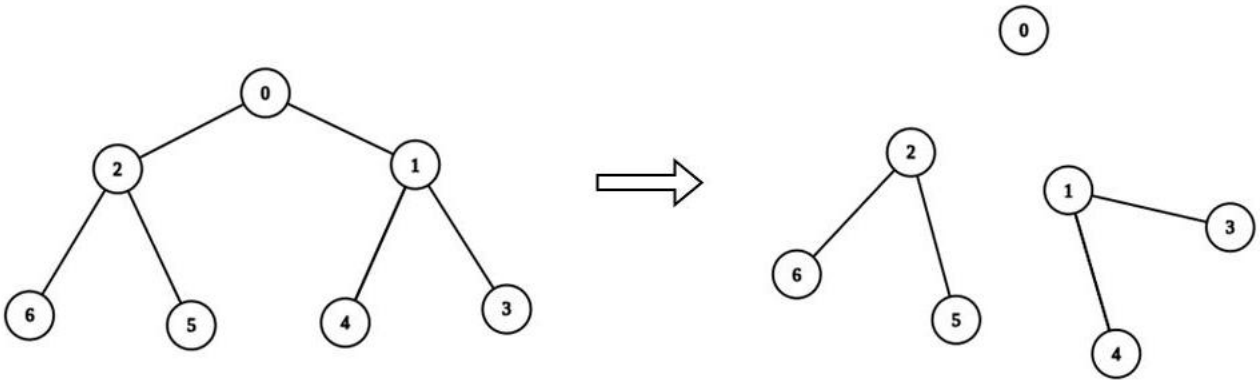
User Accepted:	76
User Tried:	98
Total Accepted:	78
Total Submissions:	129
Difficulty:	Hard

Example 1:




```
Input: n = 5, edges = [[0,2],[1,2],[1,3],[2,4]], values = [1,8,1,4,4], k = 6
Output: 2
Explanation: We remove the edge connecting node 1 with 2. The resulting split is valid because:
- The value of the component containing nodes 1 and 3 is values[1] + values[3] = 12.
- The value of the component containing nodes 0, 2, and 4 is values[0] + values[2] + values[4] = 6.
It can be shown that no other valid split has more than 2 connected components.
```

Example 2:




```
Input: n = 7, edges = [[0,1],[0,2],[1,3],[1,4],[2,5],[2,6]], values = [3,0,6,1,5,2,1], k = 3
Output: 3
Explanation: We remove the edge connecting node 0 with 2, and the edge connecting node 0 with 1. The resulting split is valid b
- The value of the component containing node 0 is values[0] = 3.
- The value of the component containing nodes 2, 5, and 6 is values[2] + values[5] + values[6] = 9.
- The value of the component containing nodes 1, 3, and 4 is values[1] + values[3] + values[4] = 6.
It can be shown that no other valid split has more than 3 connected components.
```

Constraints:

- $1 <= n <= 3 * 10^4$
- edges.length == n - 1
- edges[i].length == 2
- $0 <= a_i, b_i < n$
- values.length == n
- $0 <= values[i] <= 10^9$
- $1 <= k <= 10^9$
- Sum of values is divisible by k .
- The input is generated such that edges represents a valid tree.

JavaScript

```javascript
const initializeGraph = (n) => { let g = []; for (let i = 0; i < n; i++) { g.push([]); } return g; };
const packUG = (g, edges) => { for (const [u, v] of edges) { g[u].push(v); g[v].push(u); } };

let g, dp, vals;
const maxKDivisibleComponents = (n, edges, values, k) => {
    g = initializeGraph(n), dp = Array(n).fill(0), vals = values;
    packUG(g, edges);
    tree_dp(0, -1);
    return dp.filter(x => x % k == 0).length;
};

const tree_dp = (cur, par) => {
    let subTreeSum = vals[cur];
    for (const child of g[cur]) {
        if (child != par) {
            tree_dp(child, cur);
            subTreeSum += dp[child];
        }
    }
    dp[cur] = subTreeSum;
};
```

Custom Testcase Use Example Testcases

Run Submit

Submission Result: Accepted (/submissions/detail/1063294028/) More Details > (/submissions/detail/1063294028/)

Share your acceptance!