Cut the Tree

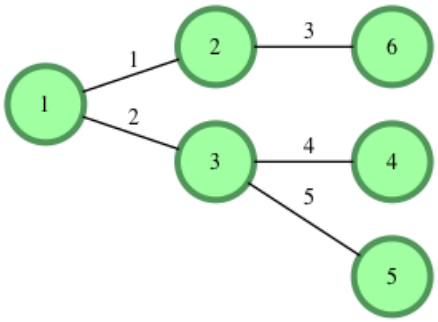
HackerRank

There is an undirected tree where each vertex is numbered from 1 to n, and each contains a data value. The *sum* of a tree is the sum of all its nodes' data values. If an edge is cut, two smaller trees are formed. The *difference* between two trees is the absolute value of the difference in their sums.

Given a tree, determine which edge to cut so that the resulting trees have a minimal *difference* between them, then return that difference.

Example
$data = [1, 2, 3, 4, 5, 6]$
$edges = [(1, 2), (1, 3), (2, 6), (3, 4), (3, 5)]$

In this case, node numbers match their weights for convenience. The graph is shown below.



The values are calculated as follows:

Edge Cut	Tree 1 Sum	Tree 2 Sum	Absolute Difference
1	8	13	5
2	9	12	3
3	6	15	9
4	4	17	13
5	5	16	11

The minimum absolute difference is 3.

Note: The given tree is *always* rooted at vertex 1.

Function Description

Complete the *cutTheTree* function in the editor below.

cutTheTree has the following parameter(s):

- *int data[n]:* an array of integers that represent node values
- *int edges[n-1][2]:* an 2 dimensional array of integer pairs where each pair represents nodes connected by the edge

Returns

- *int:* the minimum achievable absolute difference of tree sums

Input Format

The first line contains an integer n, the number of vertices in the tree.
The second line contains n space-separated integers, where each integer u denotes the $node[u]$ data value, $data[u]$.
Each of the $n-1$ subsequent lines contains two space-separated integers u and v that describe edge $u \leftrightarrow v$ in tree t.

Constraints

- $3 \leq n \leq 10^5$
- $1 \leq data[u] \leq 1001$, where $1 \leq u \leq n$.

Sample Input

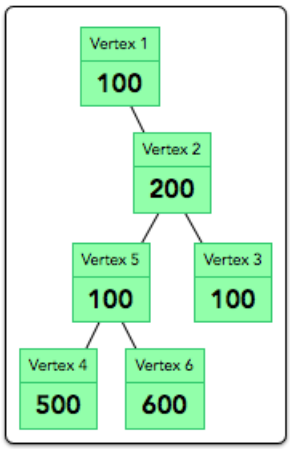
```
STDIN                       Function
-----                       --------
6                           data[] size n = 6
100 200 100 500 100 600     data = [100, 200, 100, 500, 100, 600]
1 2                         edges = [[1, 2], [2, 3], [2, 5], [4, 5], [5, 6]]
2 3
2 5
4 5
5 6
```

Sample Output

```
400
```

Explanation

We can visualize the initial, uncut tree as:



There are $n-1=5$ edges we can cut:

1. Edge $1 \leftrightarrow 2$ results in $d_{1 \leftrightarrow 2} = 1500 - 100 = 1400$
2. Edge $2 \leftrightarrow 3$ results in $d_{2 \leftrightarrow 3} = 1500 - 100 = 1400$
3. Edge $2 \leftrightarrow 5$ results in $d_{2 \leftrightarrow 5} = 1200 - 400 = 800$

4. Edge $4 \leftrightarrow 5$ results in $d_{4 \leftrightarrow 5} = 1100 - 500 = 600$
5. Edge $5 \leftrightarrow 6$ results in $d_{5 \leftrightarrow 6} = 1000 - 600 = 400$

The minimum *difference* is 400.