Problem D. Min-Max Tree

Input file:	standard input
Output file:	standard output
Time limit:	2 seconds
Memory limit:	512 megabytes

You are given an unrooted tree with n vertices. The i-th vertex has a weight a_i.

The value of a connected component in the tree is defined as the difference between the maximum weight and the minimum weight among all vertices in the component. Your task is to delete some edges of the tree to partition it into several connected components, such that the sum of the values of all connected components is maximized.

Output the maximum possible sum of values after the partition.

Input

The first line contains an integer n ($1 \le n \le 10^6$), the number of vertices in the tree.

The second line contains n integers $a_1, a_2, \ldots, a_n$ ($0 \le a_i \le 10^9$), the weights of the vertices.

Each of the next $n-1$ lines contains two integers u and v ($1 \le u, v \le n$), representing an edge between vertex u and vertex v. It is guaranteed that the given edges form a tree.

Due to large input sizes, please use fast I/O methods.

Output

Output a single integer — the maximum possible sum of values of all connected components after partitioning the tree.

Examples

standard input	standard output
8 1 3 7 5 2 8 4 6 1 2 1 3 1 4 3 5 5 6 3 7 7 8	14
7 5 9 9 6 6 3 2 3 5 4 6 2 6 5 4 7 2 4 1	13

Note

In the first test case, one of the optimal partitions is: $\{1, 2, 3, 4\}, \{5, 6\}, \{7, 8\}$**. The values of the** 3 **components are** $6, 6, 2$**, respectively.**