D. Weighting a Tree

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

You are given a connected undirected graph with n vertices and m edges. The vertices are enumerated from 1 to n.

You are given n integers $c_1, c_2, ..., c_n$, each of them is between $-n$ and n, inclusive. It is also guaranteed that the parity of c_v equals the parity of degree of vertex v. The degree of a vertex is the number of edges connected to it.

You are to write a weight between $-2 \cdot n^2$ and $2 \cdot n^2$ (inclusive) on each edge in such a way, that for each vertex v the sum of weights on edges connected to this vertex is equal to c_v, or determine that this is impossible.

Input

The first line contains two integers n and m ($2 \le n \le 10^5$, $n - 1 \le m \le 10^5$) — the number of vertices and the number of edges.

The next line contains n integers $c_1, c_2, ..., c_n$ ($-n \le c_i \le n$), where c_i is the required sum of weights of edges connected to vertex i. It is guaranteed that the parity of c_i equals the parity of degree of vertex i.

The next m lines describe edges of the graph. The i-th of these lines contains two integers a_i and b_i ($1 \le a_i, b_i \le n$; $a_i \ne b_i$), meaning that the i-th edge connects vertices a_i and b_i.

It is guaranteed that the given graph is connected and does not contain loops and multiple edges.

Output

If there is no solution, print "NO".

Otherwise print "YES" and then m lines, the i-th of them is the weight of the i-th edge w_i ($-2 \cdot n^2 \le w_i \le 2 \cdot n^2$).

Examples

input

```
3 3
2 2 2
1 2
2 3
1 3
```

output

```
YES
1
1
1
```

input

```
4 3
-1 0 2 1
1 2
2 3
3 4
```

output

```
YES
-1
1
1
```

input

```
6 6
3 5 5 5 1 5
1 4
3 2
4 3
4 5
3 5
5 6
```

output

```
YES
3
5
3
-1
-3
5
```

input

```
4 4
4 4 2 4
1 2
2 3
3 4
4 1
```

output

```
NO
```