Task C12. FUEL

⌛0.5 sec. 💾 256 MB

Author: Aleksandar Gatev

With the recent stock market crash, gas stations in Gabrovo have fallen into crisis. Kyusho regularly fills up his car in Gabrovo, but with today's increased gas prices, he has no choice but to use public transportation. Faced with such a difficult choice, Kyusho set out to fix the situation.

There are N gas stations in Gabrovo in total, numbered from 1 to N, with M bidirectional streets, each of which connecting two different gas stations. There is at most one street between each pair of gas stations. From each gas station, you can reach any other by moving along the streets. Each gas station has a fuel balance — an integer c_i (it is possible for the balance to be negative). From time to time, a gas station drains fuel from all its neighbors (those to which there is a direct street), thus increasing its balance by k_i, where k_i is the number of neighboring gas stations. In the same time, their balance decreases by one (even if it is already negative).

Kyusho knows that the people of Gabrovo will continue to pump fuel from each other until they all end up with a non-negative balance. Using his connections, he can convince each of them to rob his neighbors a certain number of times. But that's where the problem comes in — Kyusho is not sure how to end up with no gas stations with a negative balance. Help him by writing the program **fuel** that, given a map of gas stations and their balances, finds what instructions Kyusho should give.

Input

The first line of the standard input contains two natural number N and M – the number of gas stations and the number of bidirectional streets between them. The next line contains N space-separated integers c_i – the balances of the gas stations. Each of the the last M lines contains two different numbers A and B, indicating that there is a street between gas stations with numbers A and B.

Output

If no solution exists, print **"Impossible"** on the only line of standard output. Otherwise, print **"Possible"** on the first line. On the next line print N integers w_i, separated by a space – how many times each of the gas stations must drain its neighbors so that in the end they all have a non-negative balance. Your solution is considered correct if in the end each gas station has a non-negative balance and for every $1 \leq i \leq N$ it is satisfied that $0 \leq w_i \leq 10^{18}$.

Constraints

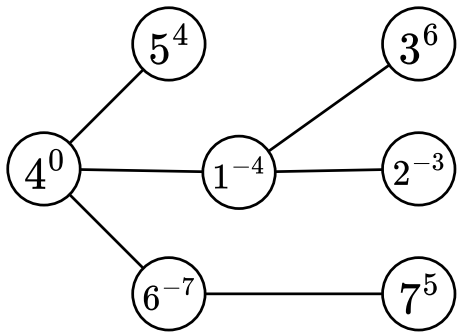
- $1 \leq N \leq 10^5$
- $N - 1 \leq M \leq N$
- $-10^4 \leq c_i \leq 10^4$
- $1 \leq A, B \leq N$, $A \neq B$
- From each gas station you can reach any other by moving along the streets.

Subtasks

Subtask	**Points**	**Required subtasks**	N	M	**Other constraints**
0	0	–	–	–	The example test cases.
1	5	–	$\leq 10^5$	$= N - 1$	All streets are between gas stations with consecutive numbers and $c_i \leq c_{i+1}$ for each $1 \leq i \leq N - 1$.
2	12	–	≤ 2000	$= N - 1$	All streets are between gas stations with consecutive numbers.
3	7	–	$\leq 10^5$	$= N - 1$	All gas stations apart from one have exactly one neighboring gas station.
4	15	–	≤ 500	$= N - 1$	–
5	12	2, 4	≤ 2000	$= N - 1$	–
6	14	1 – 5	$\leq 10^5$	$= N - 1$	–
7	16	–	≤ 2000	$\leq N$	$c_1 + c_2 + ... + c_N \neq 0$
8	13	2, 4, 5, 7	≤ 2000	$\leq N$	–
9	6	0 – 8	$\leq 10^5$	$\leq N$	–

The points for a subtask are given only if all tests for it and the required subtasks are passed ***successfully****.*

Examples

Input	Output	Explanation of the example
7 6 -4 -3 6 0 4 -7 5 2 1 1 3 1 4 4 5 4 6 6 7	Possible 6 9 0 4 0 6 1	Gas station 1 robs its neighbors 6 times, increasing its balance by $6 \times 3 = 18$, but it is also robbed $9 + 0 + 4 = 13$ times, making its final balance equal to $-4 + 18 - 13 = 1$. All other gas stations are left with zero balance. Illustration of the gas stations and initial balances: 
7 7 0 1 -1 4 -3 -1 1 1 2 2 3 3 4 4 5 5 1 1 6 4 7	Possible 4 3 3 1 4 5 0	After following Kyusho's instructions, all gas stations have a balance of 0, except for the one with number 3, which has a balance of 1.
3 3 1 0 -1 1 2 2 3 1 3	Impossible	No matter how gas stations rob each other, they will always never all have a non-negative balance.