Colors

Consider a **connected undirected graph** with N nodes and M edges. Initially every node u has a color $a[u]$, encoded by an integer between 1 and N. You can repeatedly modify node colors by assigning $a[u] = \min(a[u], a[v])$, where u and v are connected by an edge.

Given a destination coloring $b[1] \dots b[N]$, determine whether you can transform a into b.

Input data

There are several test cases per input file and you should answer each of them separately.

The first line contains the number of test cases. Each test case is structured as

```
N M
a[1] a[2] ... a[N]
b[1] b[2] ... b[N]
u1 v1
u2 v2
...
uM vM
```

Output data

For every test case you should print, on a separate line, 1 if a can be transformed into b using the above-mentioned operation and 0 otherwise.

Limits and constraints

- For all test cases, $N \le 150{,}000$ and $M \le 200{,}000$.
- For every input file, the sum of all $N \le 300{,}000$ and the sum of all $M \le 400{,}000$.
- $1 \le a[i], b[i] \le N$ for all $1 \le i \le N$.
- Time limit: 3.0 seconds
- Memory limit: 512 MB

Subtasks

Each subtask will be **scored as a group**. In order, the subtasks are:

Subtask	Points	Additional input constraints
1	15%	The graph is a **star** ($M = N - 1$ and one node is connected to every other node). The sum of N^2 among all test cases in an input file ≤ 5,000,000.
2	7%	The graph is **complete.** $N \leq 50$. The sum of $N \times M$ among all test cases in an input file ≤ 12,000,000.
3	8%	The graph is a **chain** ($M = N - 1$ and the edges form a single path). The sum of N^2 among all test cases in an input file ≤ 5,000,000.
4	15%	The graph is a **chain**, no further constraints.
5	7%	The graph is a **tree**. The sum of N^2 among all test cases in an input file ≤ 5,000,000.
6	16%	The graph is a **tree** and the coloring a is a permutation of $\{1, 2, ..., N\}$.
7	10%	The sum of $N \times M$ among all test cases in an input file ≤ 5,000,000.
8	22%	none

Example

Input	Output	Explanation
2 4 4 3 3 2 1 2 1 2 1 1 2 2 3 3 4 4 2 4 4 3 3 2 1 1 2 2 1 1 2 2 3 3 4 4 2	1 0	For the first graph, the operations needed are: $a[2] = \min(a[2], a[3]) = 2$ $a[1] = \min(a[1], a[2]) = 2$ $a[2] = \min(a[2], a[4]) = 1$