Beech Tree

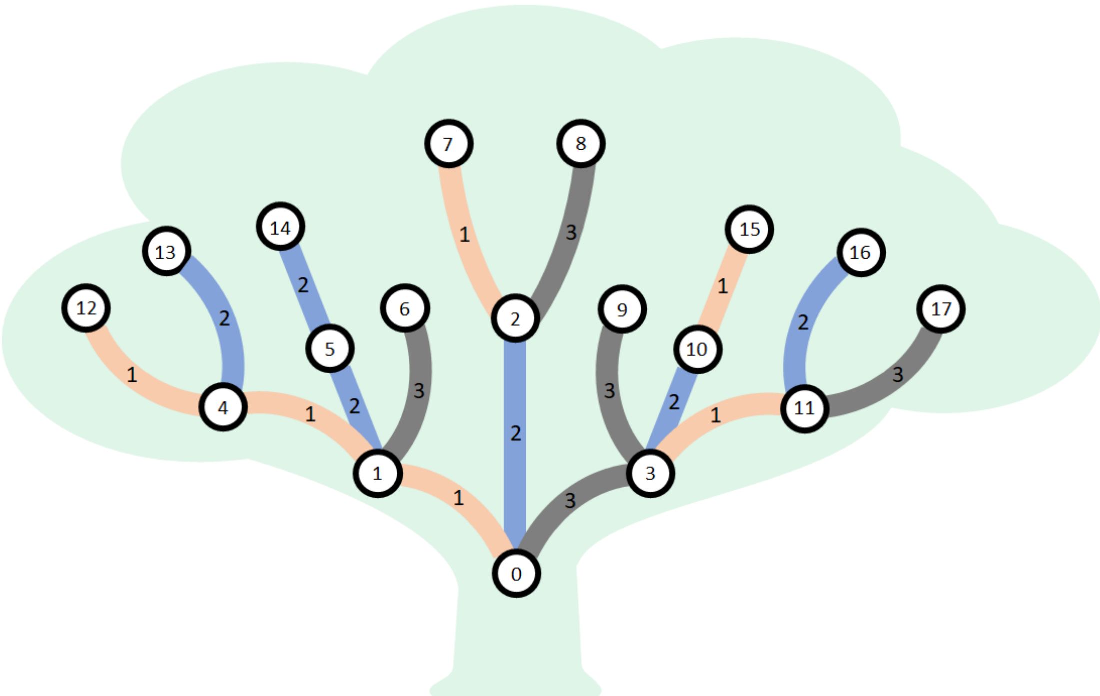
Vétyem Woods is a famous woodland with lots of colorful trees. One of the oldest and tallest beech trees is called Ős Vezér.

The tree Ős Vezér can be modeled as a set of N **nodes** and $N-1$ **edges**. Nodes are numbered from 0 to $N-1$ and edges are numbered from 1 to $N-1$. Each edge connects two distinct nodes of the tree. Specifically, edge i ($1 \leq i < N$) connects node i to node $P[i]$, where $0 \leq P[i] < i$. Node $P[i]$ is called the **parent** of node i, and node i is called a **child** of node $P[i]$.

Each edge has a color. There are M possible edge colors numbered from 1 to M. The color of edge i is $C[i]$. Different edges may have the same color.

Note that in the definitions above, the case $i = 0$ does not correspond to an edge of the tree. For convenience, we let $P[0] = -1$ and $C[0] = 0$.

For example, suppose that Ős Vezér has $N = 18$ nodes and $M = 3$ possible edge colors, with 17 edges described by connections $P = [-1, 0, 0, 0, 1, 1, 1, 2, 2, 3, 3, 3, 4, 4, 5, 10, 11, 11]$ and colors $C = [0, 1, 2, 3, 1, 2, 3, 1, 3, 3, 2, 1, 1, 2, 2, 1, 2, 3]$. The tree is displayed in the following figure:



Árpád is a talented forester who likes to study specific parts of the tree called **subtrees**. For each r such that $0 \leq r < N$, the subtree of node r is the set $T(r)$ of nodes with the following properties:

- Node r belongs to $T(r)$.
- Whenever a node x belongs to $T(r)$, all children of x also belong to $T(r)$.
- No other nodes belong to $T(r)$.

The size of the set $T(r)$ is denoted as $|T(r)|$.

Árpád recently discovered a complicated but interesting subtree property. Árpád's discovery involved a lot of playing with pen and paper, and he suspects you might need to do the same to understand it. He will also show you multiple examples you can then analyze in detail.

Suppose we have a fixed r and a permutation $v_0, v_1, \ldots, v_{|T(r)|-1}$ of the nodes in the subtree $T(r)$.

For each i such that $1 \leq i < |T(r)|$, let $f(i)$ be the number of times the color $C[v_i]$ appears in the following sequence of $i-1$ colors: $C[v_1], C[v_2], \ldots, C[v_{i-1}]$.

(Note that $f(1)$ is always 0 because the sequence of colors in its definition is empty.)

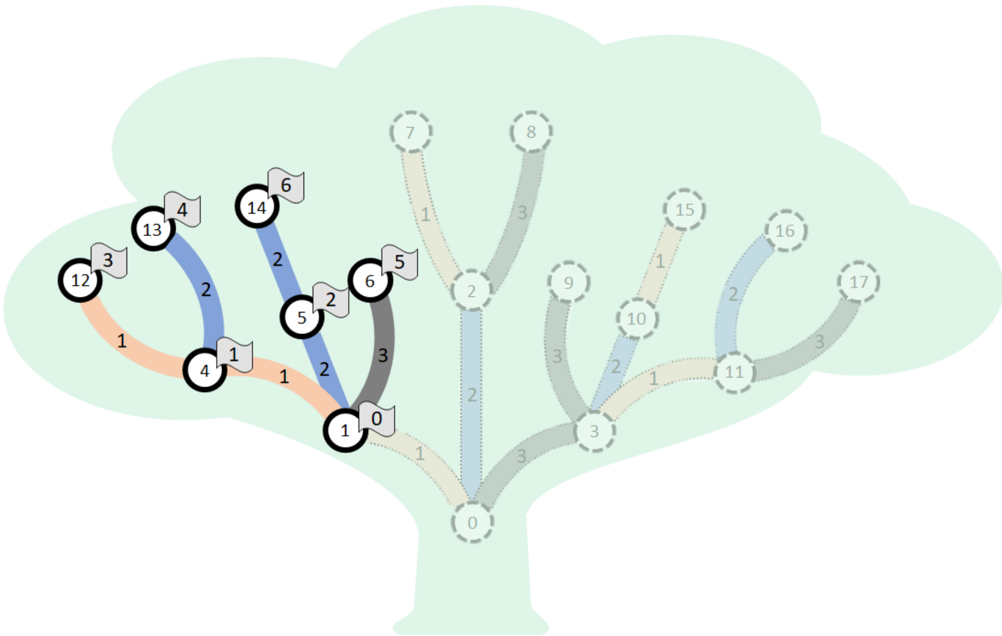
The permutation $v_0, v_1, \ldots, v_{|T(r)|-1}$ is a **beautiful permutation** if and only if all the following properties hold:

- $v_0 = r$.
- For each i such that $1 \leq i < |T(r)|$, the parent of node v_i is node $v_{f(i)}$.

For any r such that $0 \leq r < N$, the subtree $T(r)$ is a **beautiful subtree** if and only if there exists a beautiful permutation of the nodes in $T(r)$. Note that according to the definition every subtree which consists of a single node is beautiful.

Consider the example tree above. It can be shown that the subtrees $T(0)$ and $T(3)$ of this tree are not beautiful. The subtree $T(14)$ is beautiful, as it consists of a single node. Below, we will show that the subtree $T(1)$ is also beautiful.

Consider the sequence of distinct integers $[v_0, v_1, v_2, v_3, v_4, v_5, v_6] = [1, 4, 5, 12, 13, 6, 14]$. This sequence is a permutation of the nodes in $T(1)$. The figure below depicts this permutation. The labels attached to the nodes are the indices at which those nodes appear in the permutation.



We will now verify that this is a *beautiful permutation*.

- $v_0 = 1$.
- $f(1) = 0$ since $C[v_1] = C[4] = 1$ appears 0 times in the sequence $[]$.
  - Correspondingly, the parent of v_1 is v_0. That is, the parent of node 4 is node 1. (Formally, $P[4] = 1$.)
- $f(2) = 0$ since $C[v_2] = C[5] = 2$ appears 0 times in the sequence $[1]$.
  - Correspondingly, the parent of v_2 is v_0. That is, the parent of 5 is 1.
- $f(3) = 1$ since $C[v_3] = C[12] = 1$ appears 1 time in the sequence $[1, 2]$.
  - Correspondingly, the parent of v_3 is v_1. That is, the parent of 12 is 4.
- $f(4) = 1$ since $C[v_4] = C[13] = 2$ appears 1 time in the sequence $[1, 2, 1]$.
  - Correspondingly, the parent of v_4 is v_1. That is, the parent of 13 is 4.
- $f(5) = 0$ since $C[v_5] = C[6] = 3$ appears 0 times in the sequence $[1, 2, 1, 2]$.
  - Correspondingly, the parent of v_5 is v_0. That is, the parent of 6 is 1.
- $f(6) = 2$ since $C[v_6] = C[14] = 2$ appears 2 times in the sequence $[1, 2, 1, 2, 3]$.
  - Correspondingly, the parent of v_6 is v_2. That is, the parent of 14 is 5.

As we could find a *beautiful permutation* of the nodes in $T(1)$, the subtree $T(1)$ is a *beautiful subtree*.

Your task is to help Árpád decide for every subtree of Ős Vezér whether it is beautiful.

Implementation Details

You should implement the following procedure.

```
int[] beechtree(int N, int M, int[] P, int[] C)
```

- N: the number of nodes in the tree.

- M: the number of possible edge colors.
- P, C: arrays of length N describing the edges of the tree.
- This procedure should return an array b of length N. For each r such that $0 \leq r < N$, $b[r]$ should be 1 if $T(r)$ is beautiful, and 0 otherwise.
- This procedure is called exactly once for each test case.

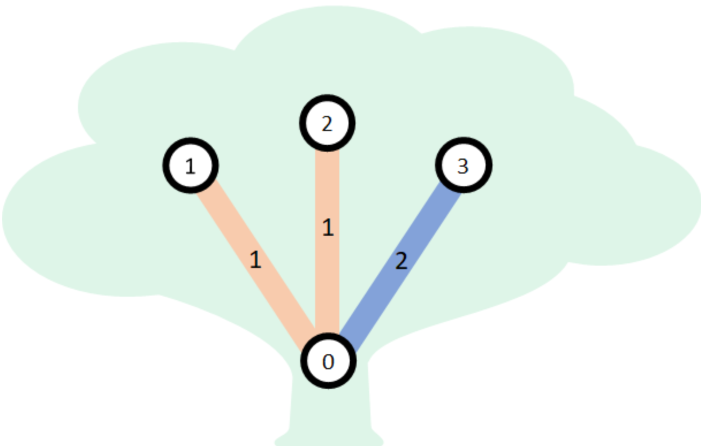
Examples

Example 1

Consider the following call:

```
beechtree(4, 2, [-1, 0, 0, 0], [0, 1, 1, 2])
```

The tree is displayed in the following figure:



$T(1)$, $T(2)$, and $T(3)$ each consist of a single node and are therefore beautiful. $T(0)$ is not beautiful. Therefore, the procedure should return $[0, 1, 1, 1]$.

Example 2

Consider the following call:

```
beechtree(18, 3,
          [-1, 0, 0, 0, 1, 1, 1, 2, 2, 3, 3, 3, 4, 4, 5, 10, 11, 11],
          [0, 1, 2, 3, 1, 2, 3, 1, 3, 3, 2, 1, 1, 2, 2, 1, 2, 3])
```

This example is illustrated in the task description above.

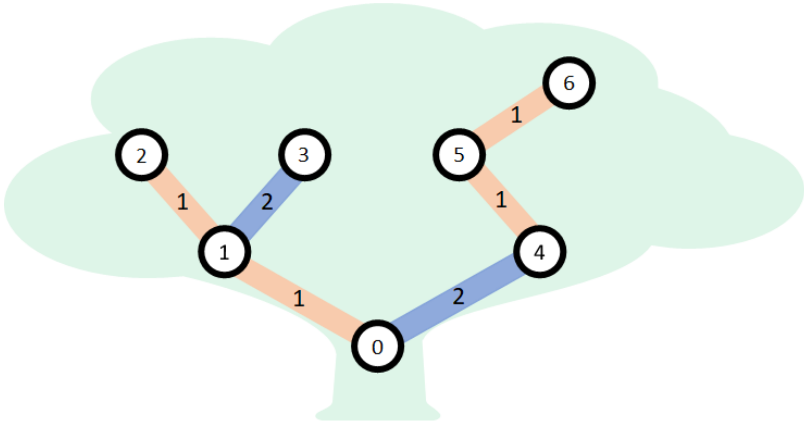
The procedure should return $[0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]$.

Example 3

Consider the following call:

```
beechtree(7, 2, [-1, 0, 1, 1, 0, 4, 5], [0, 1, 1, 2, 2, 1, 1])
```

This example is illustrated in the following figure.



$T(0)$ is the only subtree that is not beautiful. The procedure should return $[0, 1, 1, 1, 1, 1, 1]$.

Constraints

- $3 \le N \le 200\,000$
- $2 \le M \le 200\,000$
- $0 \le P[i] < i$ (for each i such that $1 \le i < N$)
- $1 \le C[i] \le M$ (for each i such that $1 \le i < N$)
- $P[0] = -1$ and $C[0] = 0$

Subtasks

1. (9 points) $N \le 8$ and $M \le 500$
2. (5 points) Edge i connects node i to node $i-1$. That is, for each i such that $1 \le i < N$, $P[i] = i - 1$.
3. (9 points) Each node other than node 0 is either connected to node 0, or is connected to a node which is connected to node 0. That is, for each i such that $1 \le i < N$, either $P[i] = 0$ or $P[P[i]] = 0$.
4. (8 points) For each c such that $1 \le c \le M$, there are at most two edges of color c.
5. (14 points) $N \le 200$ and $M \le 500$
6. (14 points) $N \le 2\,000$ and $M = 2$
7. (12 points) $N \le 2\,000$
8. (17 points) $M = 2$
9. (12 points) No additional constraints.

Sample Grader

The sample grader reads the input in the following format:

- line 1: $N \; M$
- line 2: $P[0] \; P[1] \; \ldots \; P[N-1]$
- line 3: $C[0] \; C[1] \; \ldots \; C[N-1]$

Let $b[0], \; b[1], \; \ldots$ denote the elements of the array returned by `beechtree`. The sample grader prints your answer in a single line, in the following format:

- line 1: $b[0] \; b[1] \; \ldots$