El árbol de haya

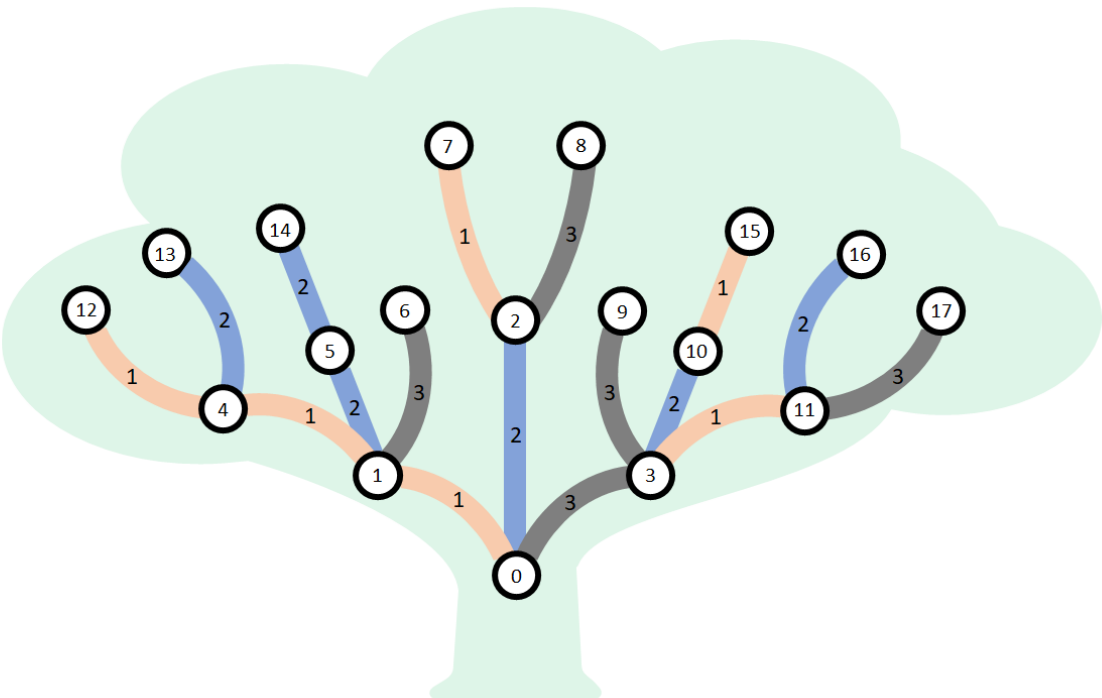
El bosque de Vétyem es un parque forestal con muchos árboles coloridos. Uno de los árboles de haya más antíguos y altos se llama Ős Vezér.

El árbol Ős Vezér puede ser modelado con un conjunto de N **nodos** y $N-1$ **aristas**. Los nodos están numerados de 0 hasta $N-1$ y las aristas están numeradas de 1 a $N-1$. Cada arista conecta dos nodos diferentes del árbol. Específicamente, la arista i ($1 \leq i < N$) conecta el nodo i al nodo $P[i]$, donde $0 \leq P[i] < i$. El nodo $P[i]$ se llama el **padre** del nodo i, y el nodo i se llama un **hijo** del nodo $P[i]$..

Cada arista tiene un color. Hay M colores de arista posibles numerados de 1 hasta M. El color de la arista i es $C[i]$. Aristas diferentes pueden tener el mismo color.

Nótese que con las definiciones anteriores, el caso $i = 0$ no corresponde a una arista del árbol. Por conveniencia, diremos que $P[0] = -1$ y $C[0] = 0$.

Por ejemplo, supongamos que Ős Vezér tiene $N = 18$ nodos y $M = 3$ colores de arista posibles, con 17 aristas descritas por las conexiones $P = [-1, 0, 0, 0, 1, 1, 1, 2, 2, 3, 3, 3, 4, 4, 5, 10, 11, 11]$ y los colores $C = [0, 1, 2, 3, 1, 2, 3, 1, 3, 3, 2, 1, 1, 2, 2, 1, 2, 3]$. El árbol se muestra en la siguiente figura:



Árpád es un silvicultor talentoso al que le gusta estudiar partes específica del árbol llamados **subárboles**. Para cada r tal que $0 \leq r < N$, el subárbol del nodo r es el conjunto de nodos $T(r)$ con las siguientes propiedades:

- El nodo r pertenece a $T(r)$.
- Cuando un nodo x pertenece a $T(r)$, todos los hijos de x pertenecen a $T(r)$.
- Ningún otro nodo pertenece a $T(r)$.

El tamaño del conjunto $T(r)$ se denota como $|T(r)|$.

Árpád ha descubierto una propiedad complicada pero interesante de los subárboles. El descubrimiento de Arpád ha requerido jugar mucho con papel y lápiz, y sospecha que puedes necesitar hacer lo mismo para entenderla. Te dará varios ejemplos que puedes analizar en detalle.

Supongamos que tenemos fijado r y una permutación $v_0, v_1, \ldots, v_{|T(r)|-1}$ de los nodos del subárbol $T(r)$.

Para cada i tal que $1 \leq i < |T(r)|$, sea $f(i)$ el número de veces que el color $C[v_i]$ aparece en la siguiente secuencia de $i-1$ colores: $C[v_1], C[v_2], \ldots, C[v_{i-1}]$.

(Nótese que $f(1)$ siempre es 0 ya que la secuencia de colores en la definición está vacía.)

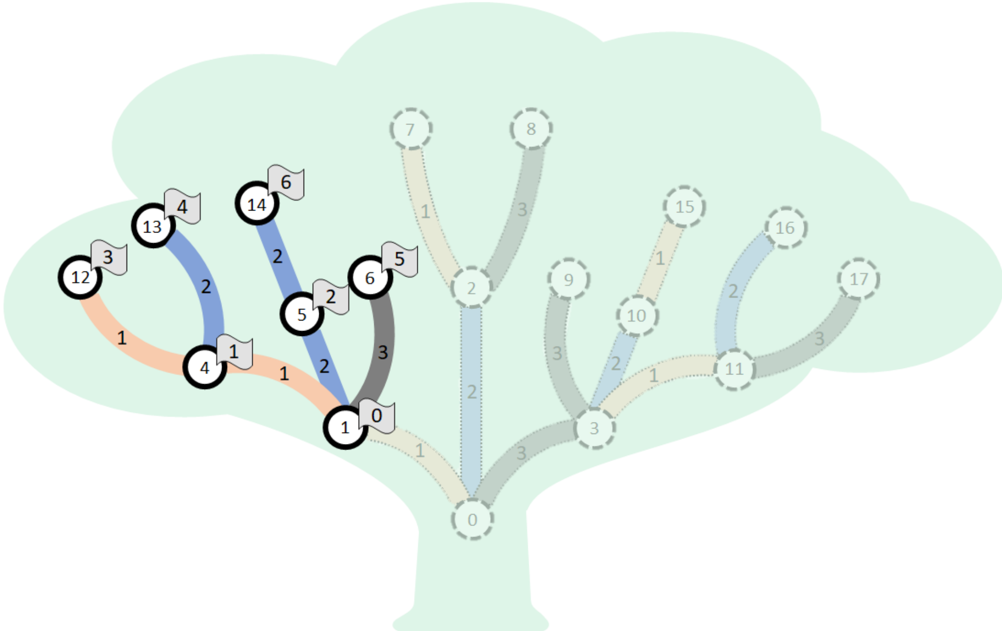
La permutación $v_0, v_1, \ldots, v_{|T(r)|-1}$ es una **permutación preciosa** si y solo si se cumplen todas las propiedades siguientes:

- $v_0 = r$.
- Para cada i tal que $1 \leq i < |T(r)|$, el padre del nodo v_i es el nodo $v_{f(i)}$.

Para cada r tal que $0 \leq r < N$, el subárbol $T(r)$ es un **subárbol precioso** si y solo si existe una permutación preciosa de los nodos en $T(r)$. Nótese que según la definición, todos los subárboles que consisten de un único nodo son preciosos.

Considera el árbol de ejemplo anterior. Se puede mostrar que los subárboles $T(0)$ y $T(3)$ de este árbol no son preciosos. El subárbol $T(14)$ es precioso, ya que contiene un único nodo. Abajo, mostraremos que el subárbol $T(1)$ también es precioso.

Considera la secuencia de enteros distintos $[v_0, v_1, v_2, v_3, v_4, v_5, v_6] = [1, 4, 5, 12, 13, 6, 14]$. Esta secuencia es una permutación de los nodos en $T(1)$. La figura siguiente muestra esta permutación. Las etiquetas junto a los nodos son los índices en los que aparecen en la permutación.



Ahora comprovaremos que es una *permutación preciosa*.

- $v_0 = 1$.
- $f(1) = 0$ ya que $C[v_1] = C[4] = 1$ aparece 0 veces en la secuencia $[]$.
  - De manera correspondiente, el padre de v_1 es v_0. Es decir, el padre del nodo 4 es el nodo 1. (Formalmente, $P[4] = 1$.)
- $f(2) = 0$ ya que $C[v_2] = C[5] = 2$ aparece 0 veces en la secuencia $[1]$.
  - De manera correspondiente, el padre de v_2 es v_0. Es decir, el padre de 5 es 1.
- $f(3) = 1$ ya que $C[v_3] = C[12] = 1$ aparece 1 veces en la secuencia $[1, 2]$.
  - De manera correspondiente, el padre de v_3 es v_1. Es decir, el padre de 12 es 4.
- $f(4) = 1$ ya que $C[v_4] = C[13] = 2$ aparece 1 veces en la secuencia $[1, 2, 1]$.
  - De manera correspondiente, el padre de v_4 es v_1. Es decir, el padre de 13 es 4.
- $f(5) = 0$ ya que $C[v_5] = C[6] = 3$ aparece 0 veces en la secuencia $[1, 2, 1, 2]$.
  - De manera correspondiente, el padre de v_5 es v_0. Es decir, el padre de 6 es 1.
- $f(6) = 2$ ya que $C[v_6] = C[14] = 2$ aparece 2 veces en la secuencia $[1, 2, 1, 2, 3]$.
  - De manera correspondiente, el padre de v_6 es v_2. Es decir, el padre de 14 es 5.

Como hemos encontrado una *permutación preciosa* de los nodos de $T(1)$, el subárbol de $T(1)$ es un *subárbol precioso*.

Tu tarea es ayudar a Árpád decidir para cada subárbol de Ős Vezér si es precioso o no.

Detalles de implementación

Debes implementar la siguiente función:

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

- N: el número de nodos del árbol.

- M: el número de colores posibles de las aristas.
- P, C: arrays de longitud N describiendo las aristas del árbol.
- La función debe devolver un array b de longitud N. Para cada r tal que $0 \leq r < N$, $b[r]$ tiene que ser 1 si $T(r)$ es precioso, y 0 en caso contrario.
- La función se llama una única vez en cada juego de pruebas.

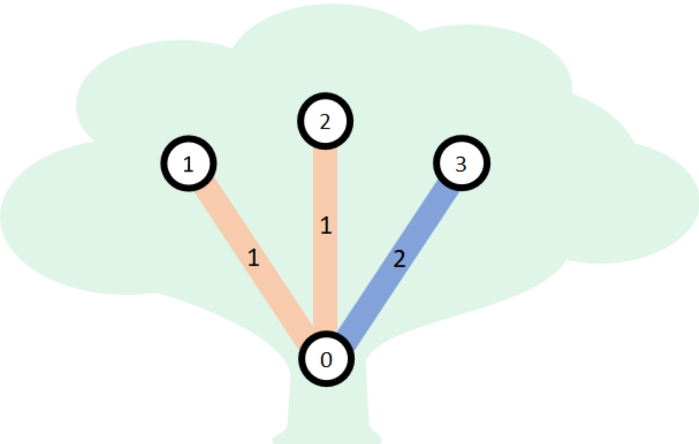
Ejemplos

Ejemplo 1

Considera la siguiente llamada:

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

El árbol se muestra en la siguiente figura:



$T(1)$, $T(2)$, y $T(3)$ cada uno contiene un solo nodo y por lo tanto son preciosos. $T(0)$ no es precioso. Por lo tanto, la función debe devolver $[0, 1, 1, 1]$.

Ejemplo 2

Considera la siguiente llamada:

```
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])
```

Este ejemplo se ha mostrado en la descripción de la tarea arriba.

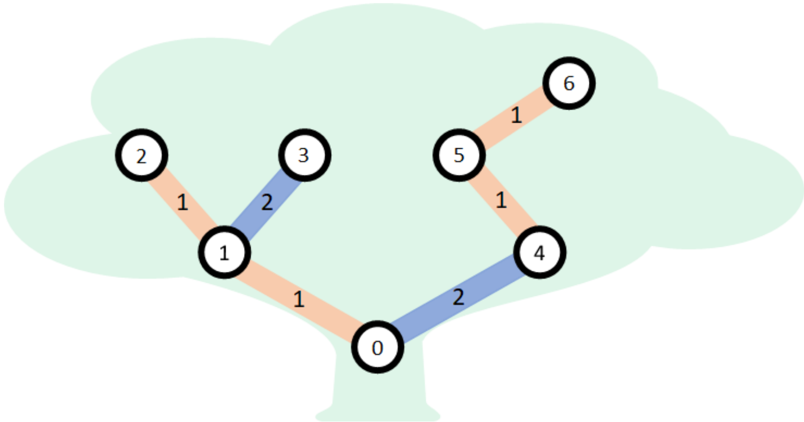
La función debe devolver $[0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]$.

Ejemplo 3

Considera la siguiente llamada:

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

Este ejemplo se muestra en la siguiente figura.



$T(0)$ es el único subárbol que no es precioso. La función debe devolver $[0, 1, 1, 1, 1, 1, 1]$.

Restricciones

- $3 \leq N \leq 200\,000$
- $2 \leq M \leq 200\,000$
- $0 \leq P[v] < v$ (para cada i tal que $1 \leq i < N$)
- $1 \leq C[v] \leq M$ (para cada i tal que $1 \leq i < N$)
- $P[0] = -1$ y $C[0] = 0$

Subtareas

1. (9 puntos) $N \leq 8$ y $M \leq 500$
2. (5 puntos) La arista i conecta el nodo i al nodo $i - 1$. Eso es, para cada$i$ tal que $1 \leq i < N$, $P[i] = i - 1$.
3. (9 puntos) Cada nodo (sin contar el nodo 0) está conectado al nodo 0 o a un nodo conectado al nodo 0. Esto es, para cada i tal que $1 \leq i < N$, o $P[i] = 0$ o $P[P[i]] = 0$.
4. (8 puntos) Para cada c tal que $1 \leq c \leq M$, hay como mucho dos aristas de color c.
5. (14 puntos) $N \leq 200$ y $M \leq 500$
6. (14 puntos) $N \leq 2\,000$ y $M = 2$
7. (12 puntos) $N \leq 2\,000$
8. (17 puntos) $M = 2$
9. (12 puntos) Sin restricciones adicionales.

Grader de ejemplo

El grader de ejemplo lee la entrada con el siguiente formato:

- línea 1: $N \ M$

- línea 2: $P[0]\ P[1]\ \ldots\ P[N-1]$
- línea 3: $C[0]\ C[1]\ \ldots\ C[N-1]$

Sea $b[0],\ b[1],\ \ldots$ denota los elementos del array devueltos por `beechtree`. El grader de ejemplo escribe tu respuesta en una sola línea, en el siguiente formato:

- línea 1: $b[0]\ b[1]\ \ldots$