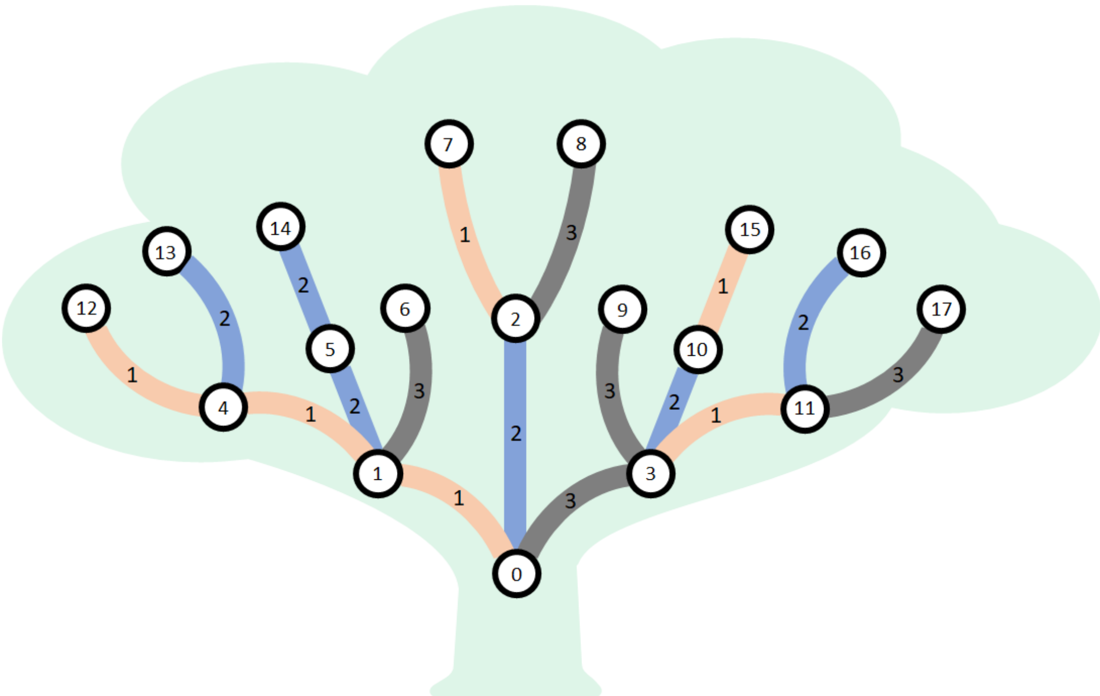
Un grande faggio

Nella foresta di Vétyem Woods vive un grande faggio, chiamato Ős Vezér.

Come sempre, questo albero è composto da N **nodi** e $N-1$ **archi**. I nodi sono numerati da 0 a $N-1$, gli archi da 1 a $N-1$. Ogni arco collega due nodi distinti dell'albero. In particolare, l'arco v ($1 \leq v < N$) collega il nodo v al nodo $P[v]$, dove $0 \leq P[v] < v$ (quindi $P[v]$ è **padre** di v e v è un **figlio** di $P[v]$).

Ogni arco ha uno di M colori, numerati da 1 a M. Il colore dell'arco v è $C[v]$. Archi diversi possono avere lo stesso colore. Inoltre, per comodità, poniamo $P[0] = -1$ e $C[0] = 0$.

Per esempio, supponiamo che Ős Vezér abbia $N = 18$ nodi ed $M = 3$ possibili colori, con 17 archi descritti dall'array $P = [-1, 0, 0, 0, 1, 1, 1, 2, 2, 3, 3, 3, 4, 4, 5, 10, 11, 11]$ e colori descritti dall'array $C = [0, 1, 2, 3, 1, 2, 3, 1, 3, 3, 2, 1, 1, 2, 2, 1, 2, 3]$. L'albero è rappresentato nella figura seguente:



Árpád è un botanico di successo specializzato nello studio di **sottoalberi**. Per ogni $0 \leq r < N$, il sottoalbero del nodo r è l'insieme $T(r)$ di nodi tale che:

- r appartiene a $T(r)$;
- se un nodo x appartiene a $T(r)$, tutti i figli di x appartengono a $T(r)$;
- nessun altro nodo appartiene a $T(r)$.

La dimensione dell'insieme $T(r)$ è denotata $|T(r)|$.

Árpád ha scoperto un'interessante (e complicata) proprietà di alcuni sottoalberi. Per trovarla, Árpád ha dovuto lavorare molto con carta e penna. Forse dovresti farlo anche tu! Per aiutarti, ti mostrerà vari esempi che potrai analizzare in dettaglio.

Per ogni $0 \leq r < N$, il sottoalbero $T(r)$ è detto **bello** se e solo se esiste una permutazione bella dei suoi nodi, definita come sotto.

Fissiamo un qualche r e una permutazione $v_0, v_1, \ldots, v_{|T(r)|-1}$ dei nodi.

Per ogni $1 \leq i < |T(r)|$, sia $f(i)$ il numero di volte che il colore $C[v_i]$ compare nella sequenza di colori $C[v_1], C[v_2], \ldots, C[v_{i-1}]$.

Nota che $f(1)$ è sempre 0 perché la sequenza di colori nella sua definizione è vuota.

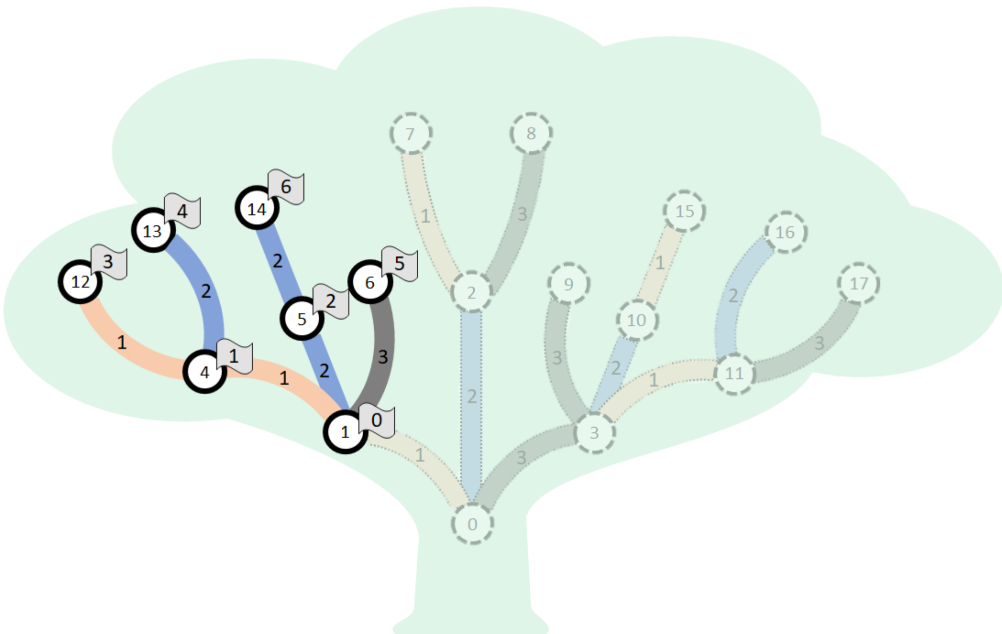
La permutazione $v_0, v_1, \ldots, v_{|T(r)|-1}$ è **bella** se e solo se tutte le seguenti proprietà sono vere:

- $v_0 = r$.
- Per ogni $1 \leq i < |T(r)|$, il padre del nodo v_i è $v_{f(i)}$.

Nota che ogni sottoalbero che contiene un singolo nodo è bello.

Torniamo all'esempio sopra. Si può dimostrare che $T(0)$ e $T(3)$ non sono belli. Il sottoalbero $T(14)$ è bello, dato che contiene un singolo nodo. Adesso mostriamo che anche il sottoalbero $T(1)$ è bello.

Considera la sequenza di interi distinti $[v_0, v_1, v_2, v_3, v_4, v_5, v_6] = [1, 4, 5, 12, 13, 6, 14]$. Questa è una permutazione di $T(1)$, come rappresentato nella figura qui sotto (l'indice di ogni nodo nella permutazione corrisponde al numero sull'etichetta di tale nodo).



Verifichiamo ora che la permutazione è bella:

- $v_0 = 1$.
- $f(1) = 0$ poiché $C[v_1] = C[4] = 1$ appare 0 volte nella sequenza $[]$, e $P[v_1] = P[4] = 1 = v_0$.
  - In effetti, il padre di v_1 è v_0 (cioè il padre di 4 è 1, ovvero $P[4] = 1$).
- $f(2) = 0$ poiché $C[v_2] = C[5] = 2$ appare 0 volte nella sequenza $[1]$, e $P[v_2] = P[5] = 1 = v_0$.
  - In effetti, il padre di v_2 è v_0 (cioè il padre di 5 è 1).
- $f(3) = 1$ poiché $C[v_3] = C[12] = 1$ appare 1 volta nella sequenza $[1, 2]$, e $P[v_3] = P[12] = 4 = v_1$.
  - In effetti, il padre di v_3 è v_1 (cioè il padre di 12 è 4).
- $f(4) = 1$ poiché $C[v_4] = C[13] = 2$ appare 1 volta nella sequenza $[1, 2, 1]$, e $P[v_4] = P[13] = 4 = v_1$.
  - In effetti, il padre di v_4 è v_1 (cioè il padre di 13 è 4).
- $f(5) = 0$ poiché $C[v_5] = C[6] = 3$ appare 0 volte nella sequenza $[1, 2, 1, 2]$, e $P[v_5] = P[6] = 1 = v_0$.
  - In effetti, il padre di v_5 è v_0 (cioè il padre di 6 è 1).
- $f(6) = 2$ poiché $C[v_6] = C[14] = 2$ appare 2 volte nella sequenza $[1, 2, 1, 2, 3]$, e $P[v_6] = P[14] = 5 = v_2$.
  - In effetti, il padre di v_3 è v_2 (cioè il padre di 14 è 5).

Avendo trovato una permutazione bella dei nodi in $T(1)$, tale sottoalbero è effettivamente bello.

Aiuta Árpád a decidere, per ogni sottoalbero di Ős Vezér, se è bello o no.

Note di implementazione

Devi implementare la seguente funzione:

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

[Nota di Traduzione: "beech tree" vuol dire faggio]

- N: il numero di nodi dell'albero.
- M: il numero di colori possibili per gli archi.
- P, C: array di lunghezza N che descrivono gli archi dell'albero.
- Questa funzione deve restituire un array b di lunghezza N. Per ogni $0 \leq r < N$, $b[r]$ deve essere 1 se $T(r)$ è bello, e 0 altrimenti.
- Questa funzione viene chiamata esattamente una volta per ogni testcase.

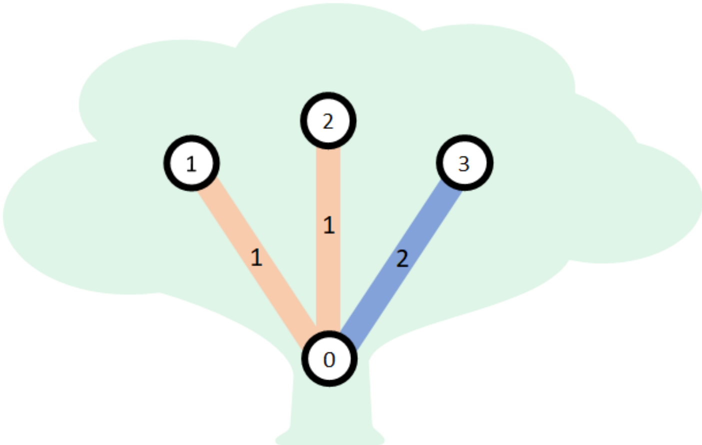
Esempi

Esempio 1

Considera la seguente chiamata:

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

L'albero è mostrato in figura:



I sottoalberi $T(1)$, $T(2)$ e $T(3)$ contengono un solo nodo e sono quindi belli. Il sottoalbero $T(0)$ non è bello. Pertanto, la funzione deve restituire $[0, 1, 1, 1]$.

Esempio 2

Considera la seguente chiamata:

```
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])
```

Questo è l'esempio nella descrizione del task.

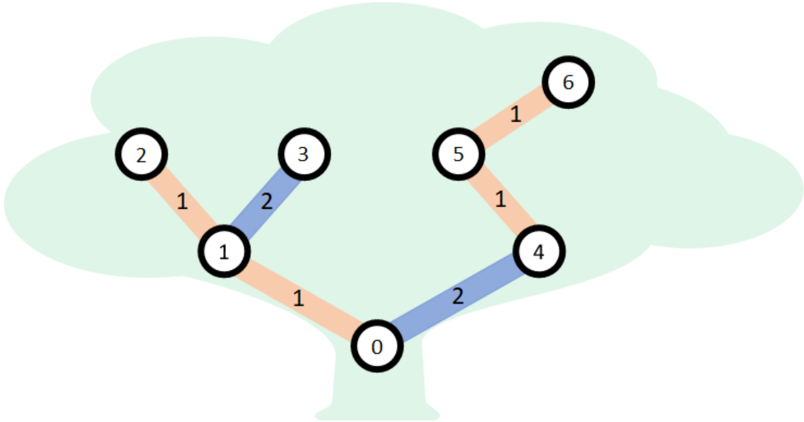
La funzione deve restituire $[0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]$.

Esempio 3

Considera la seguente chiamata:

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

Nella figura sottostante, l'unico sottoalbero non bello è $T(0)$, quindi la funzione deve restituire l'array $[0, 1, 1, 1, 1, 1, 1]$.



Assunzioni

- $3 \le N \le 200\,000$.
- $2 \le M \le 200\,000$.
- $0 \le P[i] < i$ per ogni $1 \le i < N$.
- $1 \le C[i] \le M$ per ogni $1 \le i < N$.
- $P[0] = -1$ e $C[0] = 0$.

Subtask

1. (9 punti) $N \le 8$ e $M \le 500$.
2. (5 punti) L'arco i collega il nodo i al nodo $i-1$. Cioè, per ogni $1 \le i < N$, $P[i] = i-1$.
3. (9 punti) Ogni nodo diverso da 0 è connesso direttamente a 0, oppure a un altro nodo connesso direttamente a 0. Cioè, per ogni $1 \le i < N$, $P[i] = 0$ o $P[P[i]] = 0$.
4. (8 punti) Per ogni $1 \le c \le M$, ci sono al più due archi del colore c.
5. (14 punti) $N \le 200$ e $M \le 500$.
6. (14 punti) $N \le 2\,000$ e $M = 2$.
7. (12 punti) $N \le 2\,000$.
8. (17 punti) $M = 2$.
9. (12 punti) Nessuna limitazione aggiuntiva.

Grader di esempio

Il grader di esempio legge l'input nel seguente formato:

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

Siano $b[0], b[1], \ldots$ gli elementi dell'array restituito da `beechtree`. Il grader di esempio stampa una singola riga con la tua risposta, nel seguente formato:

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