Buk

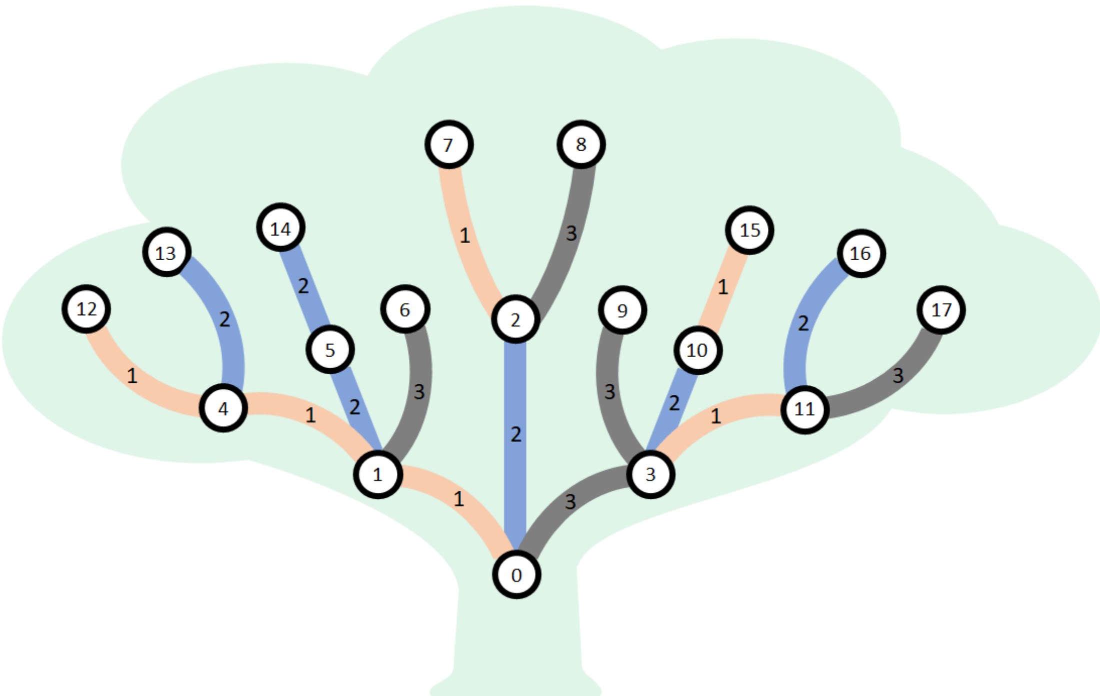
Las Vétyem to słynny teren lesisty pełen kolorowych drzew. Jeden z najstarszych i najwyższych buków nazywa się Ős Vezér.

Drzewo Ős Vezér może być przedstawione jako zbiór N **wierzchołków** oraz $N-1$ **krawędzi**. Wierzchołki są ponumerowane liczbami od 0 do $N-1$, a krawędzie są ponumerowane liczbami od 1 do $N-1$. Każda krawędź łączy dwa różne wierzchołki drzewa. Mówiąc bardziej dokładnie, krawędź i ($1 \leq i < N$) łączy wierzchołek i do wierzchołka $P[i]$, gdzie $0 \leq P[i] < v$. Wierzchołek $P[i]$ jest nazywany **rodzicem** wierzchołka i, a wierzchołek i jest nazywany **dzieckiem** wierzchołka $P[i]$.

Każda krawędź ma kolor. Jest M możliwych kolorów krawędzi, każdy z nich to liczba od 1 do M. Kolor krawędzi i to $C[i]$. Różne krawędzie mogą mieć ten sam kolor.

Zauważ, że w powyższych definicjach przypadek $i = 0$ nie odpowiada żadnej krawędzi drzewa. Dla wygody definiujemy $P[0] = -1$ oraz $C[0] = 0$.

Na przykład, niech Ős Vezér ma $N = 18$ wierzchołków oraz $M = 3$ możliwych kolorów krawędzi, a także 17 krawędzi opisanych połączeniami $P = [-1, 0, 0, 0, 1, 1, 1, 2, 2, 3, 3, 3, 4, 4, 5, 10, 11, 11]$ oraz kolorami $C = [0, 1, 2, 3, 1, 2, 3, 1, 3, 3, 2, 1, 1, 2, 2, 1, 2, 3]$. To drzewo jest przedstawione na poniższym rysunku:



Árpád jest zdolnym leśniczym, który lubi badać pewne części drzewa, które nazywamy **poddrzewami**. Dla każdego r spełniającego $0 \leq r < N$, poddrzewo wierzchołka r to zbiór $T(r)$ wierzchołków o poniższych własnościach:

- Wierzchołek r należy do $T(r)$.
- Gdy wierzchołek x należy do $T(r)$, wszystkie dzieci x także należą do $T(r)$.
- Żaden inny wierzchołek nie należy do $T(r)$.

Rozmiar zbioru $T(r)$ jest oznaczany przez $|T(r)|$.

Árpád niedawno odkrył ciekawą choć nieco skomplikowaną własność poddrzew. Odkrycie Árpáda wymagało rozpisania na kartce wielu przykładów i wydaje mu się, że być może będziesz musiał tak zrobić, aby je zrozumieć. Aby Ci to ułatwić, przygotował wiele przykładów, które możesz dokładnie przeanalizować.

Powiedzmy, że mamy ustalone r oraz permutację $v_0, v_1, \ldots, v_{|T(v)|-1}$ wierzchołków z poddrzewa $T(r)$.

Dla każdego i spełniającego $1 \leq i < |T(r)|$, niech $f(i)$ będzie liczbą wystąpień koloru $C[v_i]$ w następującym ciągu $i-1$ kolorów: $C[v_1], C[v_2], \ldots, C[v_{i-1}]$.

(Zauważ, że $f(1)$ to zawsze 0 ponieważ ciąg kolorów w jej definicji jest pusty.)

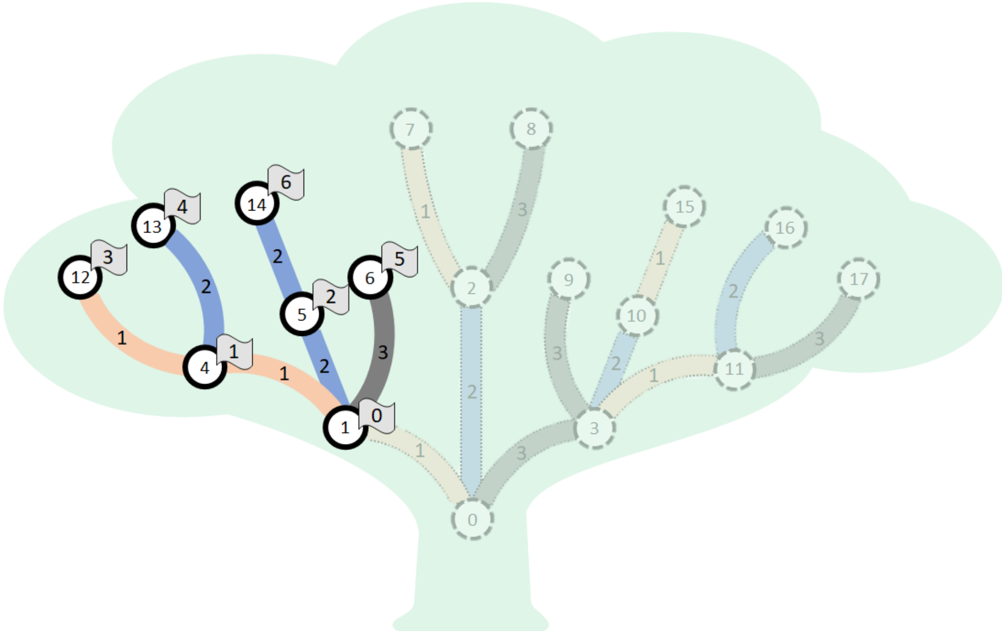
Permutacja $v_0, v_1, \ldots, v_{|T(r)|-1}$ jest **fajną permutacją** wtedy i tylko wtedy gdy są spełnione wszystkie z poniższych warunków:

- $v_0 = r$.
- Dla każdego i spełniającego $1 \leq i < |T(r)|$, rodzicem wierzchołka v_i jest wierzchołek $v_{f(i)}$.

Dla dowolnego r spełniającego $0 \leq r < N$, poddrzewo $T(r)$ jest **fajnym poddrzewem** wtedy i tylko wtedy gdy istnieje fajna permutacja wierzchołków z $T(r)$. Zauważ, że zgodnie z tą definicją każde poddrzewo złożone z jednego wierzchołka jest fajne.

Rozważ powyższe drzewo przykładowe. Można pokazać, że poddrzewa $T(0)$ oraz $T(3)$ tego drzewa nie są fajne. Poddrzewo $T(14)$ jest fajne, ponieważ składa się z pojedynczego wierzchołka. Poniżej wykażemy, że poddrzewo $T(1)$ także jest fajne.

Rozważ ciąg różnych liczb całkowitych $[v_0, v_1, v_2, v_3, v_4, v_5, v_6] = [1, 4, 5, 12, 13, 6, 14]$. Ciąg ten jest permutacją wierzchołków w $T(1)$. Poniższy rysunek przedstawia tę permutację. Etykiety przy wierzchołkach zawierają indeksy, na których wierzchołki występują w permutacji.



Sprawdzimy teraz, że jest to *fajna permutacja*.

- $v_0 = 1$.
- $f(1) = 0$ ponieważ $C[v_1] = C[4] = 1$ występuje 0 razy w ciągu $[]$.
  - Odpowiednio, rodzicem v_1 jest v_0. Czyli rodzicem wierzchołka 4 jest wierzchołek 1. (Formalnie, $P[4] = 1$.)
- $f(2) = 0$ ponieważ $C[v_2] = C[5] = 2$ występuje 0 razy w ciągu $[1]$.
  - Odpowiednio, rodzicem v_2 jest v_0. Czyli rodzicem wierzchołka 5 jest wierzchołek 1.
- $f(3) = 1$ poniewaz $C[v_3] = C[12] = 1$ występuje 1 razy w ciągu $[1, 2]$.
  - Odpowiednio, rodzicem v_3 jest v_1. Czyli rodzicem wierzchołka 12 jest wierzchołek 4.
- $f(4) = 1$ ponieważ $C[v_4] = C[13] = 2$ występuje 1 raz w ciągu $[1, 2, 1]$.
  - Odpowiednio, rodzicem v_4 jest v_1. Czyli rodzicem wierzchołka 13 jest wierzchołek 4.
- $f(5) = 0$ ponieważ $C[v_5] = C[6] = 3$ występuje 0 razy w ciągu $[1, 2, 1, 2]$.
  - Odpowiednio, rodzicem v_5 jest v_0. Czyli rodzicem wierzchołka 6 jest wierzchołek 1.
- $f(6) = 2$ ponieważ $C[v_6] = C[14] = 2$ występuje 2 razy w ciągu $[1, 2, 1, 2, 3]$.
  - Odpowiednio, rodzicem v_6 jest v_2. Czyli rodzicem wierzchołka 14 jest wierzchołek 5.

Ponieważ byliśmy w stanie znaleźć *fajną permutację* wierzchołków z $T(1)$, poddrzewo $T(1)$ jest *fajnym poddrzewem*.

Twoim zadaniem jest pomóc Árpádowi określić, dla każdego poddrzewa Ős Vezér, czy jest ono fajne.

Szczegóły implementacji

Powinieś zaimplementować następującą procedurę.

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

- N: liczba wierzchołków drzewa.
- M: liczba możliwych kolorów krawędzi.
- P, C: tablice długości N opisujące krawędzie drzewa.
- Procedura powinna zwrócić tablicę b długości N. Dla każdego r, takiego że $0 \leq r < N$, $b[r]$ powinno być równe 1, gdy $T(r)$ jest fajne, lub 0 w przeciwnym wypadku.
- Ta procedura będzie wywołana dokładnie raz dla każdego przypadku testowego.

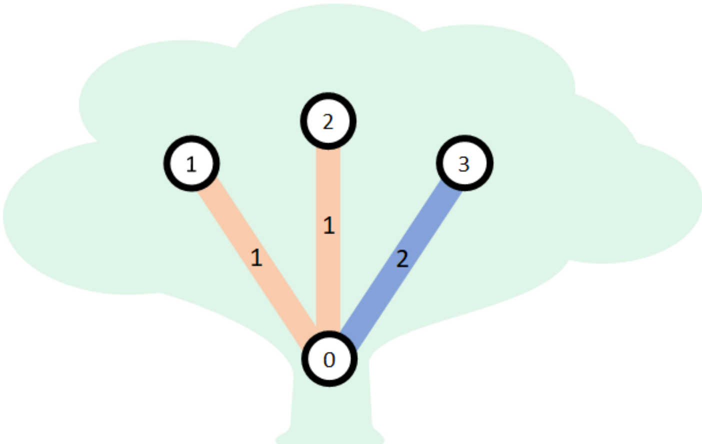
Przykłady

Przykład 1

Rozważ następujące wywołanie:

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

Drzewo jest przedstawione na poniższym rysunku:



Każde z $T(1)$, $T(2)$ i $T(3)$ jest złożone z pojedynczego wierzchołka, w związku z tym jest fajne. $T(0)$ nie jest fajne. W związku z tym procedura powinna zwrócić $[0, 1, 1, 1]$.

Przykład 2

Rozważ poniższe wywołanie:

```
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])
```

Ten przykład jest zilustrowany powyżej w treści zadania.

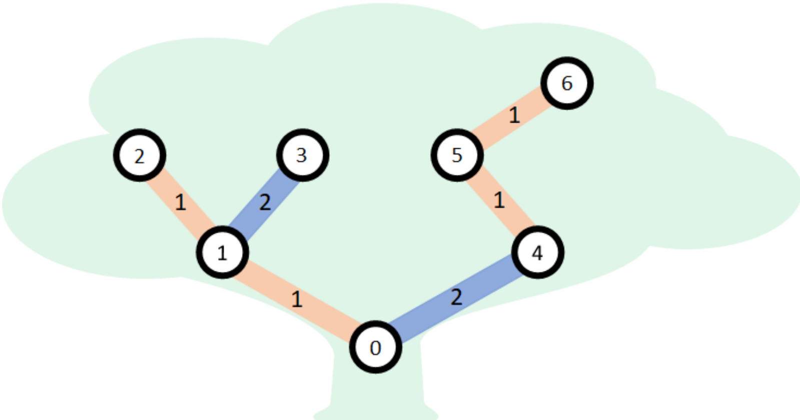
Procedura powinna zwrócić $[0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]$.

Przykład 3

Rozważ poniższe wywołanie:

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

Ten przykład jest zilustrowany na poniższym rysunku.



$T(0)$ jest jedynym poddrzewem, które nie jest fajne. Procedura powinna zwrócic $[0, 1, 1, 1, 1, 1, 1]$.

Ograniczenia

- $3 \leq N \leq 200\,000$
- $2 \leq M \leq 200\,000$
- $0 \leq P[i] < i$ (dla każdego i spełniającego $1 \leq i < N$)
- $1 \leq C[i] \leq M$ (dla każdego i spełniającego $1 \leq i < N$)
- $P[0] = -1$ oraz $C[0] = 0$

Podzadania

1. (9 punktów) $N \leq 8$ oraz $M \leq 500$
2. (5 punktów) Krawędź i łączy wierzchołek i z wierzchołkiem $i-1$. W związku z tym, dla każdego i spełniającego $1 \leq i < N$, $P[i] = i - 1$.
3. (9 punktów) Każdy wierzchołek poza wierzchołkiem 0 jest połączony z wierzchołkiem 0 lub jest połączony z wierzchołkiem połączonym z wierzchołkiem 0. Mówiąc inaczej, dla każdego i spełniającego $1 \leq i < N$, mamy $P[i] = 0$ lub $P[P[i]] = 0$.
4. (8 punktów) Dla każdego c spełniającego $1 \leq c \leq M$ mamy co najwyżej dwie krawędzie koloru c.
5. (14 punktów) $N \leq 200$ oraz $M \leq 500$
6. (14 punktów) $N \leq 2\,000$ oraz $M = 2$
7. (12 punktów) $N \leq 2\,000$
8. (17 punktów) $M = 2$
9. (12 punktów) Brak dodatkowych ograniczeń.

Przykładowy program oceniający

Przykładowy program oceniający wczytuje wejście w następującym formacie:

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

Niech $b[0],\ b[1],\ \ldots$ oznacza kolejne elementy tablicy zwróconej przez `beechtree`. Przykładowy program oceniający wypisuje Twoją odpowiedź w pojedynczym wierszu w następującym formacie:

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