Árvore Beech

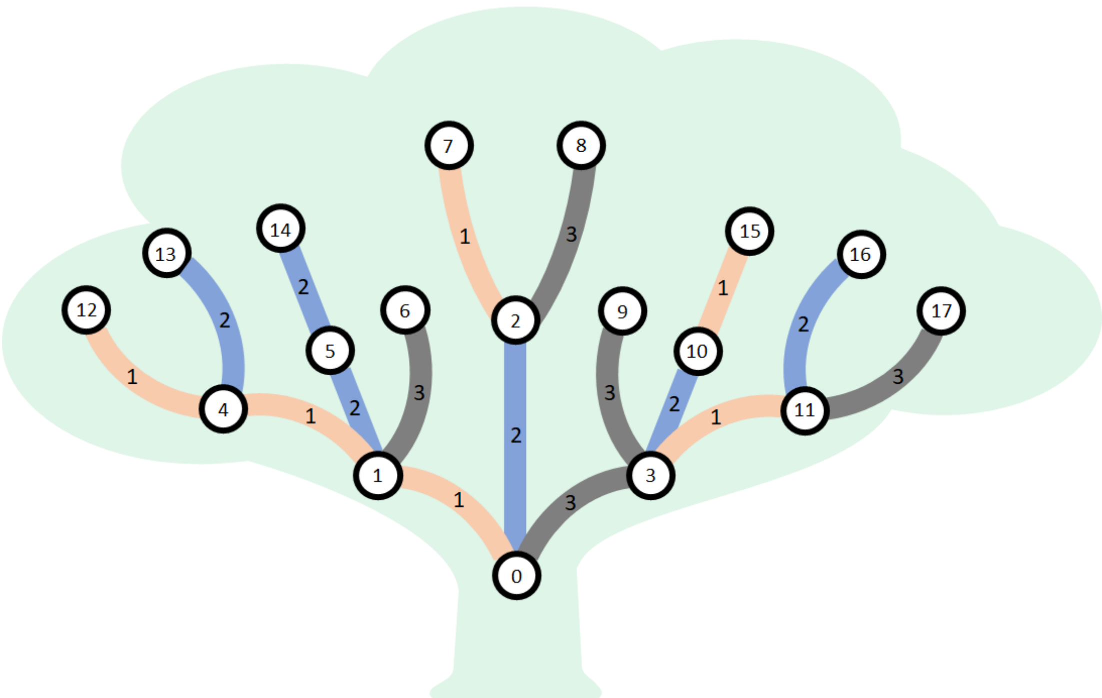
Vétyem Woods é uma famosa floresta com imensas árvores coloridas. Uma das mais altas e antigas árvores é conhecida como Ős Vezér.

A árvore Ős Vezér pode ser vista como um conjunto de N **vértices** and $N-1$ **arestas**. Os vértices são numerados de 0 a $N-1$ e as arestas numeradas de 1 a $N-1$. Cada aresta liga dois vértices distintos da árvore. Mais especificamente, a aresta i ($1 \leq i < N$) liga o vértice i ao vértice $P[i]$, onde $0 \leq P[i] < i$. O vértice $P[i]$ é chamado de **pai** do vértice i, e o vértice i é chamado de **filho** do vértice $P[i]$.

Cada aresta tem uma cor. Há M possíveis cores de aresta, numeradas de 1 a M. A cor da aresta i é $C[i]$. Diferentes arestas podem ter a mesma cor.

Note que segundo as definições acima, o caso $i = 0$ não corresponde a uma aresta da árvore. Por conveniência, deixamos $P[0] = -1$ e $C[0] = 0$.

Por exemplo, suponha que a Ős Vezér tem $N = 18$ vértices e $M = 3$ possíveis cores de aresta, com 17 arestas descritas pelas ligações $P = [-1, 0, 0, 0, 1, 1, 1, 2, 2, 3, 3, 3, 4, 4, 5, 10, 11, 11]$ e com as cores $C = [0, 1, 2, 3, 1, 2, 3, 1, 3, 3, 2, 1, 1, 2, 2, 1, 2, 3]$. A árvore pode ser visualizada na seguinte figura:



Árpád é um talentoso guarda florestal que gosta de estudar partes especificas da árvore chamadas de **subárvores**. Para cada r tal que $0 \leq r < N$, a subárvore do vértice r é o conjunto $T(r)$ de vértices com as seguintes propriedades:

- Vértice r pertence a $T(r)$.
- Sempre que um vértice x pertence a $T(r)$, todos os filhos de x também pertencem a $T(r)$.
- Nenhum outro vértice pertence a $T(r)$.

O tamanho do conjunto $T(r)$ é denotado por $|T(r)|$.

Árpád descobriu uma propriedade complicada porém interessante da subárvore. A descoberta de Árpád envolveu brincar bastante com papel e caneta, e ele suspeita que você também precise fazer o mesmo para entendê-la. Ele também vai te mostrar múltiplos exemplos que depois você pode analizar detalhadamente.

Suponha que tenhamos um r fixo e uma permutação $v_0, v_1, \ldots, v_{|T(r)|-1}$ dos vértices na subárvore $T(r)$.

Para todo i tal que $1 \leq i < |T(r)|$, seja $f(i)$ o número de vezes que a cor $C[v_i]$ aparece na seguinte sequência de $i-1$ cores: $C[v_1], C[v_2], \ldots, C[v_{i-1}]$.

(Note que $f(1)$ é sempre 0 porque a sequência de cores em sua definição é vazia.)

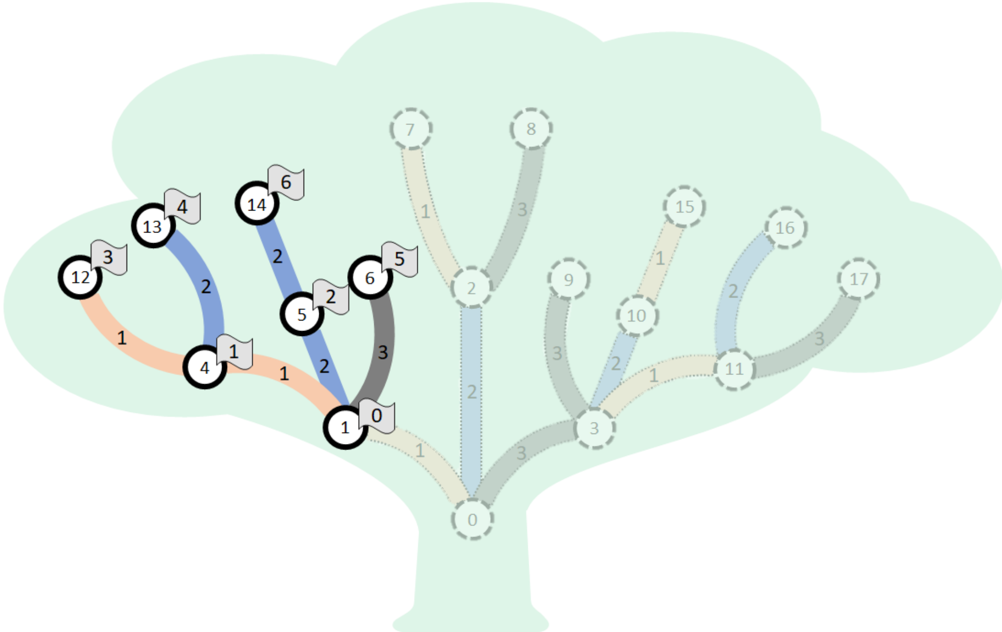
A permutação $v_0, v_1, \ldots, v_{|T(r)|-1}$ é uma **permutação linda** se e somente se satisfaz à todas as propriedades a seguir:

- $v_0 = r$.
- Para todo i tal que $1 \leq i < |T(r)|$, o pai do vértice v_i é o vértice $v_{f(i)}$.

Para qualquer r tal que $0 \leq r < N$, a subárvore $T(r)$ é uma **subárvore linda** se e somente se existe uma permutação linda dos vértices em $T(r)$. Note que de acordo com a definição toda subárvore que consiste de apenas um único vértice é linda.

Considere o exemplo da árvore anterior. Pode ser provado que as subárvores $T(0)$ e $T(3)$ desta árvore não são lindas. A subárvore $T(14)$ é linda, visto que contém um único vértice. Abaixo, mostraremos que a subárvore $T(1)$ também é linda.

Considere a sequência de inteiros distintos $[v_0, v_1, v_2, v_3, v_4, v_5, v_6] = [1, 4, 5, 12, 13, 6, 14]$. Esta sequência é uma permutação dos vértices em $T(1)$. A figura abaixo retrata essa permutação. Os rótulos anexados aos vértices são os índices nos quais estes vértices aparecem na permutação.



Vamos agora verificar que esta é uma *permutação linda*.

- $v_0 = 1$.
- $f(1) = 0$ pois $C[v_1] = C[4] = 1$ aparece 0 vez na sequência $[]$.
  - Correspondentemente, o pai de v_1 é v_0. Isto é, o pai do vértice 4 é o vértice 1. (Formalmente, $P[4] = 1$.)
- $f(2) = 0$ pois $C[v_2] = C[5] = 2$ aparece 0 vez na sequência $[1]$.
  - Correspondentemente, o pai de v_2 é v_0. Isto é, o pai do vértice 5 é o vértice 1.
- $f(3) = 1$ pois $C[v_3] = C[12] = 1$ aparece 1 vez na sequência $[1, 2]$.
  - Correspondentemente, o pai de v_3 é v_1. Isto é, o pai do vértice 12 é o vértice 4.
- $f(4) = 1$ pois $C[v_4] = C[13] = 2$ aparece 1 vez na sequência $[1, 2, 1]$.
  - Correspondentemente, o pai de v_4 é v_1. Isto é, o pai do vértice 13 é o vértice 4.
- $f(5) = 0$ pois $C[v_5] = C[6] = 3$ aparece 0 vez na sequência $[1, 2, 1, 2]$.
  - Correspondentemente, o pai de v_5 é v_0. Isto é, o pai do vértice 6 é o vértice 1.
- $f(6) = 2$ pois $C[v_6] = C[14] = 2$ aparece 2 vezes na sequência $[1, 2, 1, 2, 3]$.
  - Correspondentemente, o pai de v_6 é v_2. Isto é, o pai do vértice 14 é o vértice 5.

Como encontramos uma *permutação linda* dos vértices em $T(1)$, então a subárvore $T(1)$ é uma *subárvore linda*.

Sua tarefa é ajudar Árpád a decidir para cada subárvore da Ős Vezér se ela é linda.

Detalhes de Implementação

Você deve implementar o seguinte procedimento.

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

- N: o número de vértices da árvore.

- M: o número de possíveis cores de aresta.
- P, C: vetores de tamanho N descrevendo as arestas da árvore.
- Este procedimento deve retornar um vetor b de tamanho N. Para cada r tal que $0 \leq r < N$, $b[r]$ deve ser 1 se $T(r)$ é linda ou 0 caso contrário.
- Este procedimento é chamado exatamente uma vez para cada caso de teste.

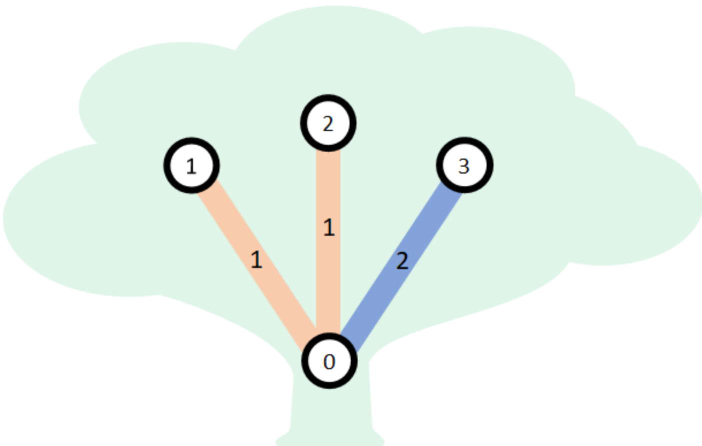
Exemplos

Exemplo 1

Considere a seguinte chamada:

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

A árvore é mostrada na figura a seguir:



$T(1)$, $T(2)$ e $T(3)$ contêm cada uma um único vértice e portanto são lindas. $T(0)$ não é linda. Então, o procedimento deve retornar $[0, 1, 1, 1]$.

Exemplo 2

Considere a seguinte chamada:

```
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 exemplo foi ilustrado no enunciado.

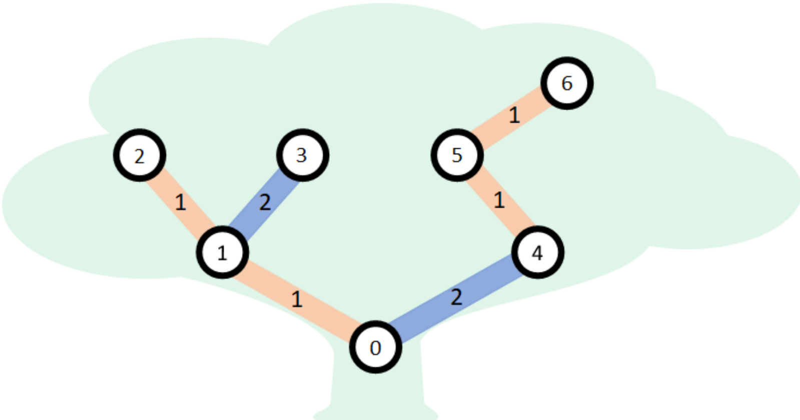
O procedimento deve retornar $[0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]$.

Exemplo 3

Considere a seguinte chamada:

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

Este exemplo é ilustrado na figura a seguir:



$T(0)$ é a única subárvore que não é linda. O procedimento deve retornar $[0, 1, 1, 1, 1, 1, 1]$.

Restrições

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

Subtarefas

1. (9 pontos) $N \leq 8$ e $M \leq 500$
2. (5 points) A aresta i liga o vértice i ao vértice $i - 1$. Isto é, para cada i tal que $1 \leq i < N$, $P[i] = i - 1$.
3. (9 pontos) Com exceção ao vértice 0, todos os vértices estão ou ligados ao vértice 0, ou ligados a um vértice que está ligado ao vértice 0. Isto é, para cada i tal que $1 \leq i < N$, vale $P[i] = 0$ or $P[P[i]] = 0$.
4. (8 pontos) Para cada c tal que $1 \leq c \leq M$, existem no máximo duas arestas com a cor c.
5. (14 pontos) $N \leq 200$ e $M \leq 500$
6. (14 pontos) $N \leq 2\,000$ e $M = 2$
7. (12 pontos) $N \leq 2\,000$
8. (17 pontos) $M = 2$
9. (12 pontos) Nenhuma restrição adicional.

Corretor exemplo

O corretor exemplo lê a entrada no seguinte formato:

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

Sejam $b[0], \ b[1], \ \ldots$ os elementos do vetor retornado por `beechtree`. O corretor exemplo imprime a sua resposta em uma única linha, no seguinte formato:

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