Bukev

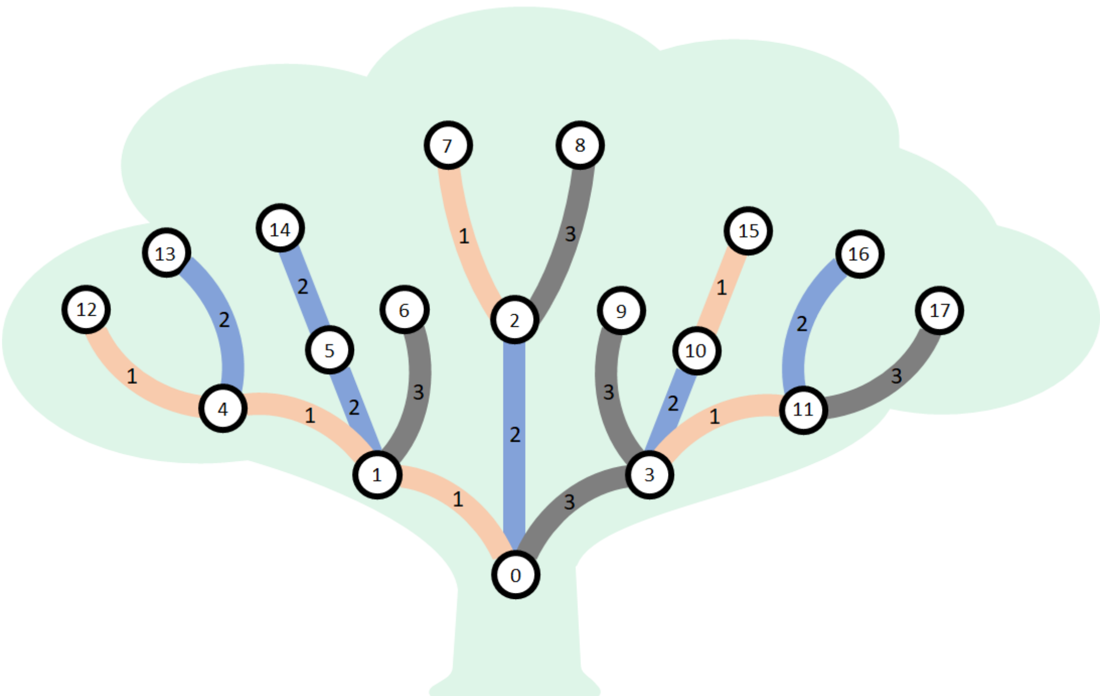
Vétyem gozd je znan po številnih pisanih drevesih. Eno najstarejših in najvišjih bukovih dreves se imenuje Ős Vezér.

Drevo Ős Vezér lahko predstavimo kot množico N **vozlišč** in $N-1$ **povezav**. Vozlišča so oštevilčena od 0 do $N-1$, povezave pa od 1 do $N-1$. Vsaka povezava povezuje dve različni vozlišči drevesa. Natančneje, povezava i ($1 \leq i < N$) povezuje vozlišče i z vozliščem $P[i]$, kjer je $0 \leq P[i] < i$. Vozlišče $P[i]$ se imenuje **starš** vozlišča i, vozlišče i pa se imenuje **otrok** vozlišča $P[i]$.

Vsaka povezava ima barvo. Obstaja M različnih barv povezav, oštevilčenih od 1 do M. Barva povezave i je $C[i]$. Različne povezave imajo lahko enako barvo.

Upoštevajte, da v zgornjih definicijah primer $i = 0$ ne ustreza nobeni povezavi drevesa. Iz praktičnih razlogov določimo, da je $P[0] = -1$ in $C[0] = 0$.

Za primer vzemimo, da ima Ős Vezér $N = 18$ vozlišč in $M = 3$ možnih barv povezav, s 17 povezavami, opisanimi s povezavami $P = [-1, 0, 0, 0, 1, 1, 1, 2, 2, 3, 3, 3, 4, 4, 5, 10, 11, 11]$ in barvami $C = [0, 1, 2, 3, 1, 2, 3, 1, 3, 3, 2, 1, 1, 2, 2, 1, 2, 3]$. Drevo je prikazano na spodnji sliki:



Árpád je nadarjen gozdar in rad preučuje posebne dele drevesa imenovane **poddrevesa**. Za vsak r, kjer $0 \leq r < N$, je poddrevo vozlišča r množica $T(r)$ vozlišč z naslednjimi lastnostmi:

- Vozlišče r pripada $T(r)$.
- Kadarkoli vozlišče x pripada $T(r)$, vsi otroci vozlišča x prav tako pripadajo $T(r)$.
- Nobeno drugo vozlišče ne pripada $T(r)$.

Moč množice $T(r)$ označimo s $|T(r)|$.

Arpád je nedavno odkril kompleksno, a zanimivo lastnost poddrevesa. Arpádovo odkritje je vključevalo veliko igranja s svinčnikom in papirjem, in sumi, da boste morali storiti enako, da bi ga razumeli. Pokazal vam bo tudi več primerov, ki jih lahko nato podrobno analizirate.

Predpostavimo, da imamo nespremenljiv r in permutacijo $v_0, v_1, \ldots, v_{|T(r)|-1}$ vozlišč v poddrevesu $T(r)$.

Za vsak i, kjer $1 \leq i < |T(r)|$, naj bo $f(i)$ število pojavitev barve $C[v_i]$ v naslednjem zaporedju $i-1$ barv: $C[v_1], C[v_2], \ldots, C[v_{i-1}]$.

(Opomba: $f(1)$ je vedno 0, ker je zaporedje barv v njegovi definiciji prazno.)

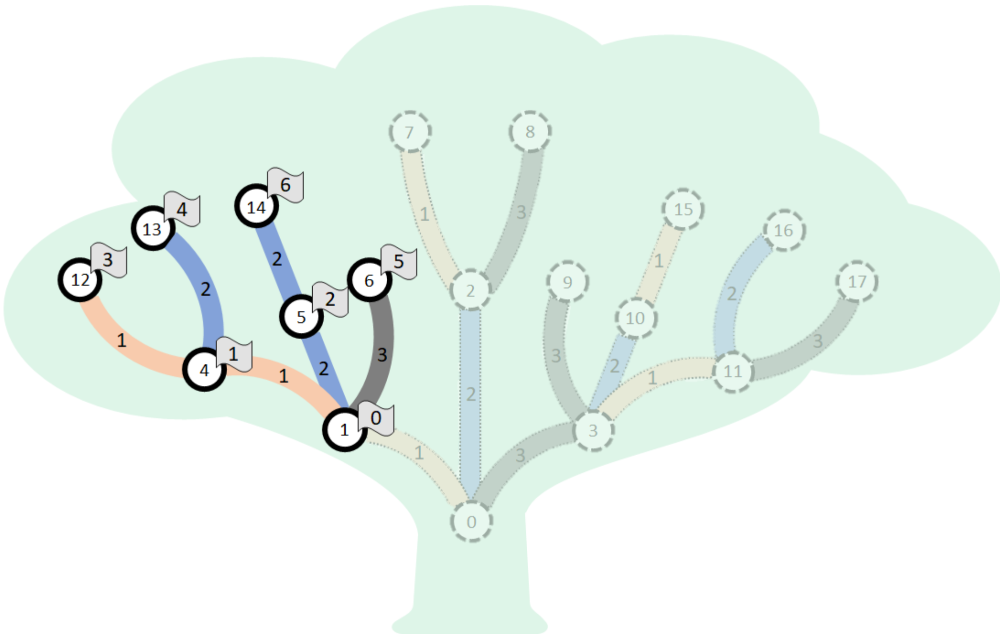
Permutacija $v_0, v_1, \ldots, v_{|T(r)|-1}$ je **lepa permutacija** natanko takrat, ko veljajo naslednje lastnosti:

- $v_0 = r$.
- Za vsak i, kjer $1 \leq i < |T(r)|$, je starš vozlišča v_i vozlišče $v_{f(i)}$.

Za vsak r, kjer $0 \leq r < N$, je poddrevo $T(r)$ **lepo poddrevo** natanko takrat, ko obstaja lepa permutacija vozlišč v $T(r)$. Upoštevajte, da je po definiciji vsako poddrevo, ki se sestoji iz enega samega vozlišča, lepo.

Upoštevajte zgornje drevo. Lahko pokažemo, da poddrevesi $T(0)$ in $T(3)$ tega drevesa nista lepi. Poddrevo $T(14)$ je lepo, saj se sestoji iz enega samega vozlišča. Spodaj bomo pokazali, da je tudi poddrevo $T(1)$ lepo.

Upoštevajte zaporedje različnih celih števil $[v_0, v_1, v_2, v_3, v_4, v_5, v_6] = [1, 4, 5, 12, 13, 6, 14]$. To zaporedje je permutacija vozlišč v $T(1)$. Spodnja slika prikazuje to permutacijo. Oznake na vozliščih so kazalci, na katerih se ta vozlišča pojavljajo v permutaciji.



Zdaj bomo preverili, ali je to *lepa permutacija*.

- $v_0 = 1$.
- $f(1) = 0$, saj se $C[v_1] = C[4] = 1$ pojavi 0-krat v zaporedju $[]$.
  - Posledično je v_0 starš v_1. To pomeni, da je starš vozlišča 4 vozlišče 1. (Formalno, $P[4] = 1$.)
- $f(2) = 0$, saj se $C[v_2] = C[5] = 2$ pojavi 0-krat v zaporedju $[1]$.
  - Posledično je v_0 starš v_2. To pomeni, da je starš vozlišča 5 vozlišče 1.
- $f(3) = 1$, saj se $C[v_3] = C[12] = 1$ pojavi enkrat v zaporedju $[1, 2]$.
  - Posledično je v_1 starš v_3. To pomeni, da je starš vozlišča 12 vozlišče 4.
- $f(4) = 1$, saj se $C[v_4] = C[13] = 2$ pojavi enkrat v zaporedju $[1, 2, 1]$.
  - Posledično je v_1 starš v_4. To pomeni, da je starš vozlišča 13 vozlišče 4.
- $f(5) = 0$, saj se $C[v_5] = C[6] = 3$ pojavi 0-krat v zaporedju $[1, 2, 1, 2]$.
  - Posledično je v_0 starš v_5. To pomeni, da je starš vozlišča 6 vozlišče 1.
- $f(6) = 2$, saj se $C[v_6] = C[14] = 2$ pojavi dvakrat v zaporedju $[1, 2, 1, 2, 3]$.
  - Posledično je v_2 starš v_6. To pomeni, da je starš vozlišča 14 vozlišče 5.

Ker smo našli *lepo permutacijo* vozlišč v poddrevesu T(1), je poddrevo T(1) *lepo poddrevo*.

Vaša naloga je pomagati Árpádu določiti za vsako poddrevo Ős Vezérja, ali je lepo.

Podrobnosti implementacije

Implementirajte funkcijo.

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

- N: število vozlišč v drevesu.
- M: število možnih barv povezav.

- P, C: polji dolžine N, ki opisujeta povezave drevesa.
- Funkcija naj vrne polje b dolžine N. Za vsak r, kjer $0 \leq r < N$, naj bo $b[r] = 1$, če je $T(r)$ lepo; sicer naj bo $b[r] = 0$.
- Funkcijo se kliče natanko enkrat za vsak testni primer.

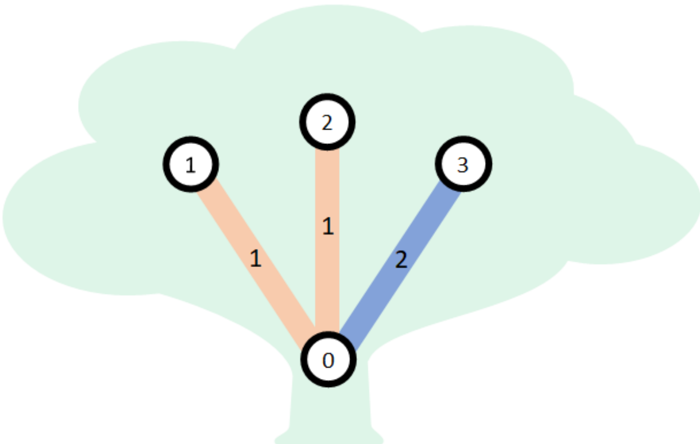
Primeri

Primer 1

Upoštevajte naslednji klic:

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

Drevo je prikazano na spodnji sliki:



$T(1)$, $T(2)$ in $T(3)$ vsebujejo po eno samo vozlišče in so zatorej lepa. $T(0)$ ni lep. Zato funkcija vrne $[0, 1, 1, 1]$.

Primer 2

Upoštevajte naslednji klic:

```
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])
```

Ta primer je prikazan v zgornjem opisu naloge.

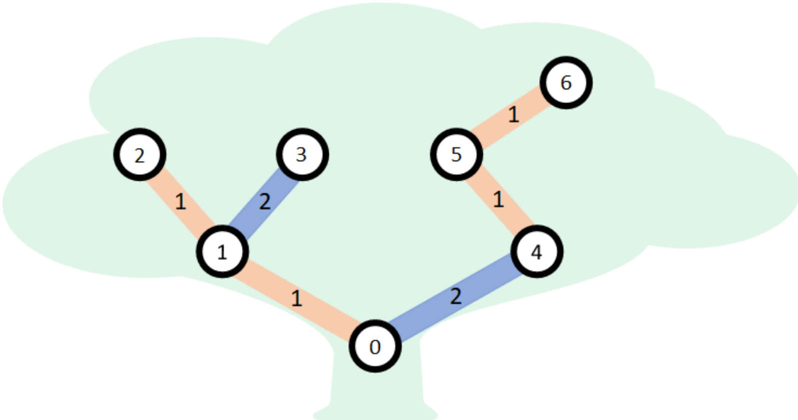
Funkcija naj vrne $[0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]$.

Primer 3

Upoštevajte naslednji klic:

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

Ta primer je prikazan na spodnji sliki.



$T(0)$ je edino poddrevo ki ni lepo. Funkcija vrne $[0,1,1,1,1,1,1]$.

Omejitve

- $3 \leq N \leq 200\,000$
- $2 \leq M \leq 200\,000$
- $0 \leq P[i] < i$ (za vsak i, kjer $1 \leq i < N$)
- $1 \leq C[i] \leq M$ (za vsak i, kjer $1 \leq i < N$)
- $P[0] = -1$ in $C[0] = 0$

Podnaloge

1. (9 točk) $N \leq 8$ in $M \leq 500$
2. (5 točk) Povezava v povezuje vozlišče i z vozliščem $i-1$. To pomeni, da za vsak i, kjer $1 \leq i < N$, velja: $P[i] = i-1$.
3. (9 točk) Vsako vozlišče, izvzemši vozlišče 0, je bodisi povezano z vozliščem 0, bodisi je povezano z vozliščem, ki je povezano z vozliščem 0. To pomeni, da za vsak i, kjer $1 \leq i < N$, velja: P[i]=0 ali P[P[i]]=0.
4. (8 točk) Za vsak c, kjer $1 \leq c \leq M$, obstajata največ dve povezavi barve c.
5. (14 točk) $N \leq 200$ in $M \leq 500$
6. (14 točk) $N \leq 2\,000$ in $M = 2$
7. (12 točk) $N \leq 2\,000$
8. (17 točk) $M = 2$
9. (12 točk) Brez dodatnih omejitev.

Vzorčni ocenjevalnik

Vzorčni ocenjevalnik bere vhod naslednje oblike:

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

Naj bodo $b[0],\ b[1],\ \ldots$ elementi polja, ki ga vrne `beechtree`. Vzorčni ocenjevalnik izpiše vaš odgovor v eni vrstici naslednje oblike:

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