Бук дарагы

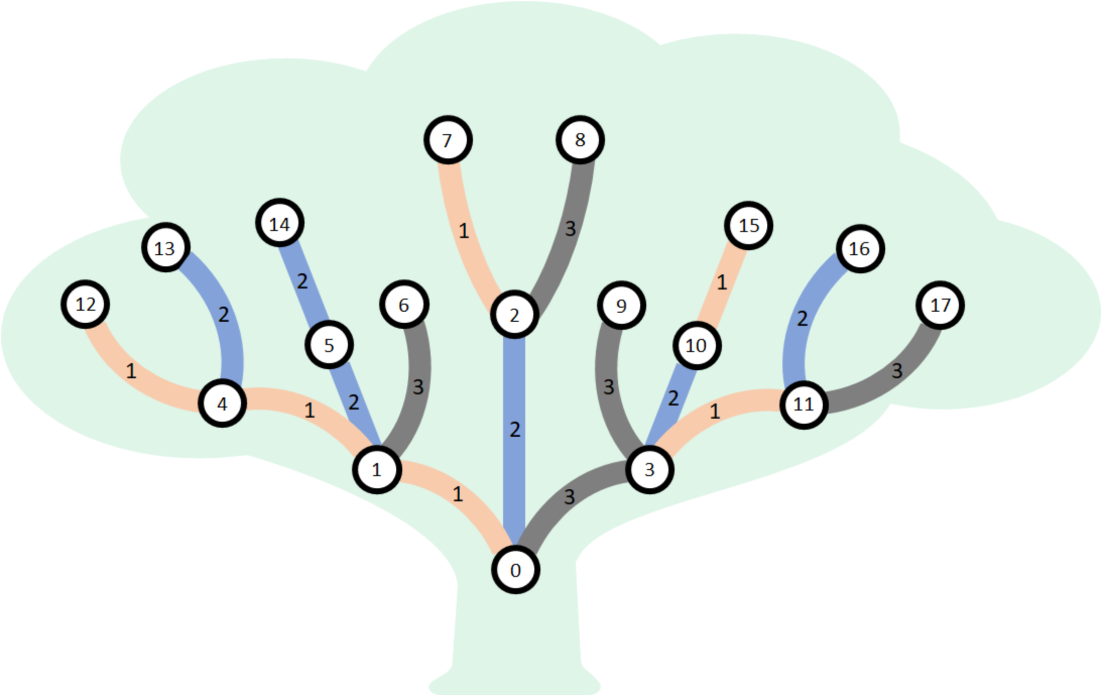
Vétyem Woods түстүү бак-дарактары көп атактуу токой болуп саналат. Эң эски жана эң бийик бук дарактарынын бири Ős Vezér деп аталат.

Ős Vezér дарагын N **түйүндөрдүн** жана $N-1$ **четтеринин** жыйындысы катары моделдештирүүгө болот. Түйүндөр 0дөн $N-1$ге чейин, ал эми четтери 1ден $N-1$ге чейин номерленген. Ар бир чети дарактын эки айырмаланган түйүндөрүн бириктирет. Тактап айтканда, i - чети ($1 \leq i < N$) i түйүнүн $P[i]$ - түйүнүнө туташтырат, мында $0 \leq P[i] < i$. $P[i]$ түйүнү i түйүнүнүн **атасы** деп аталат, ал эми i түйүнү $P[i]$ түйүнүнүн **баласы** деп аталат.

Ар бир четинин түсү бар. 1ден Mге чейин номерленген M болушу мүмкүн болгон четки түстөр бар. i четинин түсү $C[i]$. Ар кандай четтери бирдей түстө болушу мүмкүн.

Жогорудагы аныктамаларда $i=0$ учуру дарактын четине туура келбей турганын эске алыңыз. Ыңгайлуу болуу үчүн, биз $P[0]=-1$ жана $C[0]=0$ уруксат беребиз.

Мисалы, Ős Vezérде $N=18$ түйүндөрү жана $M=3$ мүмкүн болгон четтеринин түстөрү бар дейли, 17 четтери $P=[-1,0,0,0,1,1,1,2,2,3,3,3,4,4,5,10,11,11]$ жана түстөр $C=[0,1,2,3,1,2,3,1,3,3,2,1,1,2,2,1,2,3]$. Дарак төмөнкү сүрөттө көрсөтүлгөн:



Árpád таланттуу токойчу, ал дарактын **ички дарак** деп аталган айрым бөлүктөрүн изилдөөнү жакшы көрөт. $0 \leq r < N$ болгон ар бир r үчүн, r түйүнүнүн ички дарагы төмөнкү касиеттерге ээ болгон түйүндөрдүн топтому $T(r)$ болуп саналат:

- r түйүнү $T(r)$га таандык.
- Качан x түйүнү $T(r)$га таандык болсо, xтын бардык балдары да $T(r)$га таандык.
- Башка түйүндөр $T(r)$га таандык эмес.

$T(r)$ топтомунун өлчөмү $|T(r)|$ катары белгиленет.

Árpád жакында татаал, бирок кызыктуу ички дарактын касиетин тапты. Árpádдын ачылышы калем жана кагаз менен көп ойногонду камтыган жана аны түшүнүү үчүн сиз да ушундай кылышыңыз керек деп шектенүүдө. Ал ошондой эле сизге бир нече мисалдарды көрсөтөт, анда сиз майда-чүйдөсүнө чейин талдай аласыз.

Бизде $T(r)$ ички дарагындагы түйүндөрдүн туруктуу r жана алмаштыруу $v_0, v_1, \ldots, v_{|T(r)|-1}$ бар дейли.

$1 \leq i < |T(r)|$ болгон ар бир i үчүн, $f(i)$ $C[v_i]$ түсүнүн $i-1$ түстөрүнүн төмөнкү ырааттуулугунда канча жолу пайда болгондугу болсун: $C[v_1], C[v_2], \ldots, C[v_{i-1}]$.

($f(1)$ дайыма 0 экенин эске алыңыз, анткени анын аныктамасындагы түстөрдүн тизмеги бош.)

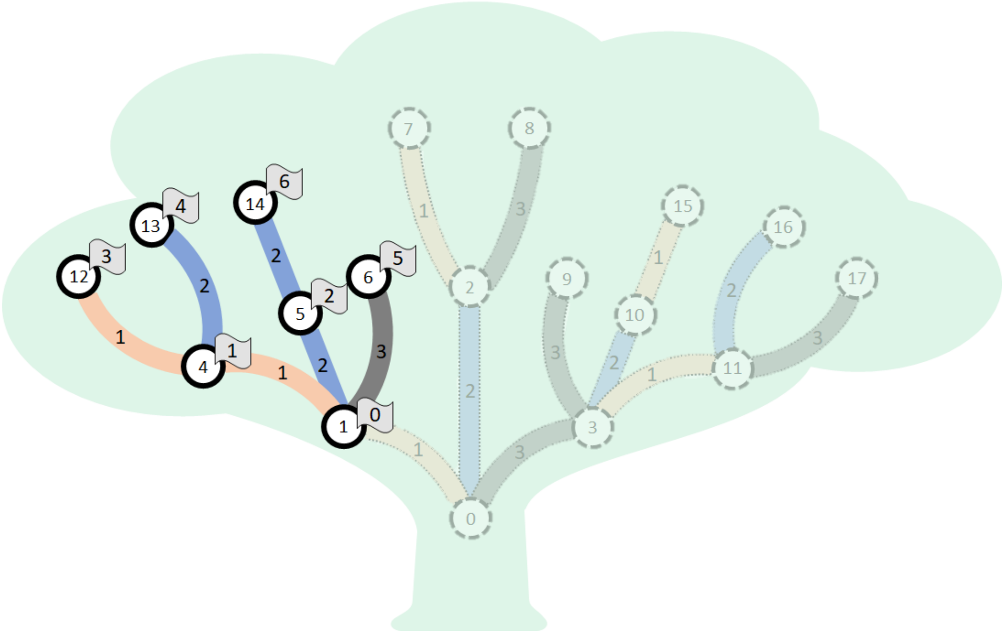
$v_0, v_1, \ldots, v_{|T(r)|-1}$ алмаштыруу төмөнкү касиеттердин бардыгы сакталган учурда гана **сонун алмаштыруу** болуп саналат:

- $v_0 = r$.
- Ар бир i үчүн $1 \leq i < |T(r)|$, v_i түйүнүнүн ата-энеси $v_{f(i)}$ түйүнү болуп саналат.

$0 \leq r < N$ болгон ар кандай r үчүн, $T(r)$ ички дарагы $T(r)$ ичинде түйүндөрдүн кооз алмаштыруусу болгондо гана **сонун ички дарак** болуп саналат. Көңүл буруңуз, аныктама боюнча бир түйүндөн турган ар бир ички дарак сонун.

Жогорудагы мисал даракты карап көрөлү. Бул дарактын $T(0)$ жана $T(3)$ ички дарактары сонун эмес экенин көрсөтсө болот. $T(14)$ ички дарагы сонун, анткени ал бир түйүндөн турат. Төмөндө биз $T(1)$ ички дарагы да сонун экенин көрсөтөбүз.

$[v_0, v_1, v_2, v_3, v_4, v_5, v_6] = [1, 4, 5, 12, 13, 6, 14]$ айырмаланган бүтүн сандардын ырааттуулугун карап көрөлү. Бул ырааттуулук $T(1)$ ичиндеги түйүндөрдүн алмаштыруусу. Төмөндөгү сүрөт бул алмаштырууну сүрөттөйт. Түйүндөргө тиркелген энбелгилер бул түйүндөр алмаштырууда пайда болгон индекстер.



Эми бул *сонун алмаштыруу* экенин текшеребиз.

- $v_0 = 1$.
- $f(1) = 0$, анткени $C[v_1] = C[4] = 1$ $[]$ ыраттуулугунда 0 жолу пайда болот.
  - Тиешелүү түрдө, v_1дын атасы v_0 болуп саналат. Башкача айтканда, 4 түйүнүнүн атасы 1 түйүнү. (Формалдуу түрдө $P[4] = 1$.)
- $f(2) = 0$, анткени $C[v_2] = C[5] = 2$ $[1]$ ыраттуулугунда 0 жолу пайда болот.
  - Тиешелүү түрдө, v_2дын атасы v_0 болуп саналат. Башкача айтканда, 5дын атасы 1.
- $f(3) = 1$, анткени $C[v_3] = C[12] = 1$ 1 жолу $[1, 2]$ ыраттуулугунда пайда болот.
  - Тиешелүү түрдө, v_3дын атасы v_1 болуп саналат. Башкача айтканда, 12нин атасы 4.
- $f(4) = 1$, анткени $C[v_4] = C[13] = 2$ 1 жолу $[1, 2, 1]$ ыраттуулугунда пайда болот.
  - Тиешелүү түрдө, v_4дын атасы v_1 болуп саналат. Башкача айтканда, 13түн атасы 4.
- $f(5) = 0$, анткени $C[v_5] = C[6] = 3$ $[1, 2, 1, 2]$ ыраатта 0 жолу пайда болот.
  - Тиешелүү түрдө, v_5тин атасы v_0 болуп саналат. Башкача айтканда, 6нын атасы 1.
- $f(6) = 2$, анткени $C[v_6] = C[14] = 2$ $[1, 2, 1, 2, 3]$ ыраттуулугунда 2 жолу пайда болот.
  - Тиешелүү түрдө, v_6нын атасы v_2 болуп саналат. Башкача айтканда, 14түн атасы 5.

Биз $T(1)$ ичинде түйүндөрдүн *сонун алмашуусун* таба алганыбыздай, $T(1)$ ички дарагы *сонун ички дарак*.

Сиздин милдет Árpád Ős Vezér ар бир ички дарак үчүн анын сонун же жокпу, чечүүгө жардам берүү болуп саналат.

Ишке ашыруу деталдары

Сиз төмөнкү процедураны ишке ашырууңуз керек.

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

- N: дарактагы түйүндөрдүн саны.
- M: мүмкүн болгон чек түстөрүнүн саны.
- P, C: дарактын четтерин сүрөттөгөн N узундуктагы массивдер.
- Бул процедура N узундуктагы b массивин кайтарышы керек. Ар бир r үчүн $0 \leq r < N$, $T(r)$ сонун болсо, $b[r]$ 1, башка учурда 0 болушу керек.
- Бул процедура ар бир тест учуру үчүн бир жолу чакырылат.

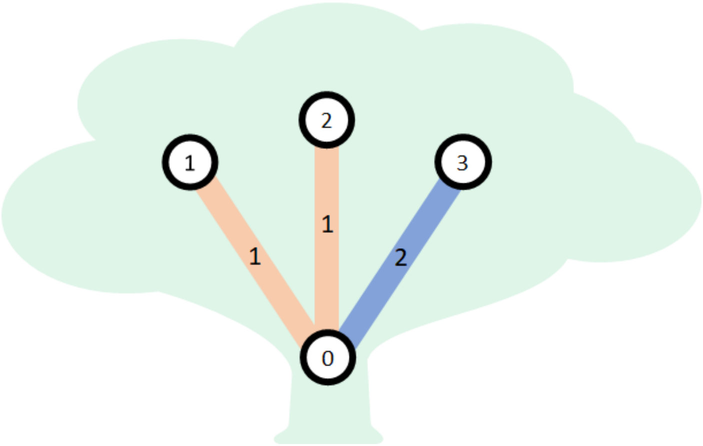
Мисалдар

1-мисал

Төмөнкү чакырууну карап көрөлү:

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

Дарак төмөнкү сүрөттө көрсөтүлгөн:



$T(1)$, $T(2)$ жана $T(3)$ ар бири бир түйүндөн турат жана ошондуктан сонун, $T(0)$ сонун эмес. Ошондуктан, процедура $[0, 1, 1, 1]$ кайтарышы керек.

2-мисал

Төмөнкү чакырууну карап көрөлү:

```
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])
```

Бул мисал жогорудагы тапшырманын сүрөттөмөсүндө көрсөтүлгөн.

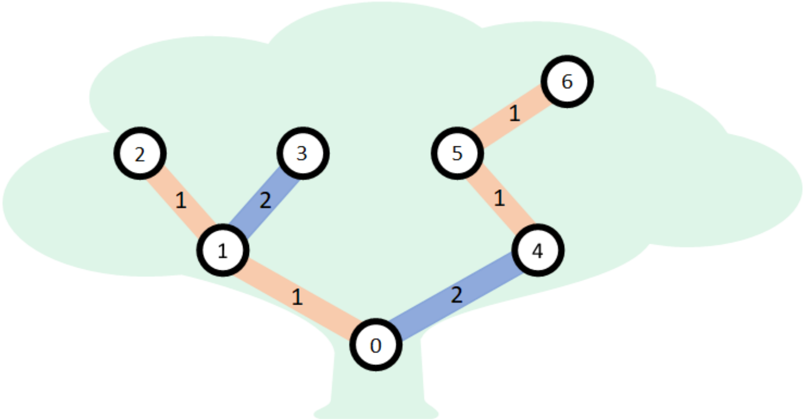
Процедура $[0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]$ кайтарып бериши керек.

3-мисал

Төмөнкү чакырууну карап көрөлү:

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

Бул мисал төмөнкү сүрөттө көрсөтүлгөн.



$T(0)$ сонун эмес жападан жалгыз дарак. Процедура $[0, 1, 1, 1, 1, 1, 1]$ кайтарышы керек.

Чектөөлөр

- $3 \leq N \leq 200\,000$
- $2 \leq M \leq 200\,000$
- $0 \leq P[i] < i$ (ар бир i үчүн $1 \leq i < N$)
- $1 \leq C[i] \leq M$ (ар бир i үчүн $1 \leq i < N$)
- $P[0] = -1$ жана $C[0] = 0$

Кошумча тапшырмалар

1. (9 упай) $N \leq 8$ жана $M \leq 500$
2. (5 упай) i - чети i- түйүн менен $i-1$- түйүнүнө туташтырат. Башкача айтканда, ар бир i үчүн $1 \leq i < N$, $P[i] = i - 1$.
3. (9 упай) 0 түйүнүнөн башка ар бир түйүн же 0- түйүнүнө туташкан, же 0- түйүнүнө туташкан түйүнгө туташкан. Башкача айтканда, ар бир i үчүн $1 \leq i < N$, же $P[i] = 0$ же $P[P[i]] = 0$.
4. (8 упай) ар бир c үчүн $1 \leq c \leq M$, c түсүнүн эң көп дегенде эки чети болот.
5. (14 упай) $N \leq 200$ жана $M \leq 500$
6. (14 упай) $N \leq 2\,000$ жана $M = 2$
7. (12 упай) $N \leq 2\,000$
8. (17 упай) $M = 2$
9. (12 упай) Кошумча чектөөлөр жок.

Үлгү Грейдер

Үлгү грейдер киргизүүнү төмөнкү форматта окуйт:

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

$b[0],\ b[1],\ \ldots$ `beechtree` тарабынан кайтарылган массивдин элементтерин билдирет. Үлгү грейдер жообуңузду төмөнкү форматта бир сапта басып чыгарат:

- 1 - сап: $b[0]\ b[1]\ \ldots$