

Kalitlar

Arxitektor Timoti yangi o'yin o'ylab topdi. Bu o'yinda n ta xona bor, xonalar 0 dan $n - 1$ gacha raqamlangan. O'yin boshlanganida har bir xonada aynan bir donadan kalit bor. Har bir kalitning turi bor va turlar 0 dan $n - 1$ gacha bo'lishi mumkin. i - xonadagi kalitning turi $r[i]$. Nazarda tutingki, har xil xonalarda turgan kalitlar bir xil turda bo'lishi ham mumkin, ya'ni $r[i]$ lar har doim ham har xil emas.

Bundan tashqari, o'yinda m ta **ikki yo'nalishli** yo'laklar ham bor, ular 0 dan $m - 1$ gacha raqamlangan. Yo'lak j ($0 \leq j \leq m - 1$) har xil $u[j]$ va $v[j]$ xonalar juftligini o'zaro bog'laydi. Ikkita xona bir nechta har xil yo'laklar yordamida bog'langan bo'lishi ham mumkin.

O'yin xonalar bo'ylab yurib kalitlar terib yuradigan bir o'yinchi tomonidan o'ynaladi. O'yinchi j -yo'lakdan **yurdi** deyiladi, qachonki u $u[j]$ dan $v[j]$ ga yo'nalishida, yoki teskari yo'nalishda yursa. O'yinchi j -yo'lakdan faqatgina $c[j]$ turidagi kalitni olganidan keyingina yura oladi.

O'yinning har qanday vaqtida, o'yinchi qandaydir x xonada va ikki xil harakat amalga oshira oladi:

- x -xonadagi $r[x]$ turdagi kalitni ol (agar u oldinroq olinmagan bo'lsa)
- j -yo'lak bo'ylab yur, bu yerda yoki $u[j] = x$ yoki $v[j] = x$ shart bajarilishi kerak va $c[j]$ turdagi kalitni o'yinchi oldinroq olib qo'ygan bo'lishi kerak.

Qayd etish joizki, o'yinchi olgan kalitlarini **hech qachon** tashlab yubora olmaydi.

O'yinchi o'yinni qaysidir s xonadan qo'lida hech qanday kalitlarsiz bo'lmagan holda boshlaydi. s xonadan t -xonaga **bora olinadigan** hisoblanadi, agar o'yinchi o'yinni s dan boshlagan o'yinchi teparoqda qayd etilgan harakatlarning qandaydir ketma-ketligini bajargan holda t gacha yetib bora olsa.

Har bir i ($0 \leq i \leq n - 1$) xona uchun, i -xonadan bora olinadigan xonalar soniga $p[i]$ deyilsin. Timoti shunaqangi i lar, ya'ni indekslar to'plamini topishni hohlardiki, ularning $p[i]$ lari qiymatlari hamma $0 \leq i \leq n - 1$ lar orasida eng kichiklari bo'lsin.

Implementatsiya tafsilotlari

Siz quyidagini implementatsiya qilishingiz kerak

```
int[] find_reachable(int[] r, int[] u, int[] v, int[] c)
```

- r : uzunligi n bo'lgan massiv. Har bir i ($0 \leq i \leq n - 1$) uchun, i -xonadagi kalit turi $r[i]$.
- u, v : uzunligi m bo'lgan ikkita massiv. Har bir j ($0 \leq j \leq m - 1$) uchun, j -yo'lak $u[j]$ va $v[j]$ larni o'zaro bog'laydi.

- c : uzunligi m bo'lgan massiv. Har bir j ($0 \leq j \leq m - 1$) uchun, j -yo'lakdan yurish uchun kerak bo'ladigan kalit turi $c[j]$.
- Bu funksiya uzunligi n bo'lga sa massiv qaytarishi kerak. Har bir $0 \leq i \leq n - 1$ uchun, $sa[i]$ qiymati 1 bo'lishi kerak agar har bir j $0 \leq j \leq n - 1$ uchun $p[i] \leq p[j]$ shart bajarilsa. Aks holda, $sa[i]$ qiymati 0 bo'lishi kerak.

Misollar

Misol 1

Quyidagi funksiya chaqiruvini qarang:

```
find_reachable([0, 1, 1, 2],
               [0, 0, 1, 1, 3], [1, 2, 2, 3, 1], [0, 0, 1, 0, 2])
```

Agar o'yinchi 0-xonadan boshlasa, u quyidagi harakatlar ketma ketligini bajara oladi.

Hozirgi xona	Harakat
0	0-turdagi kalitni ol
0	0-yo'lak yordamida 1-xonaga bor
1	1-turdagi kalitni ol
1	2-yo'lak yordamida 2-xonaga bor
2	2-yo'lak yordamida 1-xonaga bor
1	3-yo'lak yordamida 3-xonaga bor

Shunday qilib, 3-xonaga 0-xonadan borish mumkin. Shunga o'xshash tarzda, biz 0 dan turib qolgan hamma xonalarga borish mumkinligini ko'rsatuvchi harakatlar ketma ketligini qurishimiz mumkin, bu anglatadiki $p[0] = 4$.

Quyidagi jadval har bir xona qolgan qaysi xonalarga bora olishini ko'rsatadi.

Boshlang'ich xona i	Bora olinadigan xonalar	$p[i]$
0	[0, 1, 2, 3]	4
1	[1, 2]	2
2	[1, 2]	2
3	[1, 2, 3]	3

Hamma xonalar orasida eng kichkina $p[i]$ qiymati 2ga teng, va bu $i = 1$ or $i = 2$ yordamida amalga oshirila olinadi. Shunday qilib, bu funksiya [0, 1, 1, 0] massivni javob sifatida qaytarishi kerak.

Misol 2

```
find_reachable([0, 1, 1, 2, 2, 1, 2],  
               [0, 0, 1, 1, 2, 3, 3, 4, 4, 5],  
               [1, 2, 2, 3, 3, 4, 5, 5, 6, 6],  
               [0, 0, 1, 0, 0, 1, 2, 0, 2, 1])
```

Quyidagi jadval har bir xonadan bora olinadigan xonalarni ko'rsatadi:

Boshlang'ich xona i	Bora olinadigan xonalar	$p[i]$
0	[0, 1, 2, 3, 4, 5, 6]	7
1	[1, 2]	2
2	[1, 2]	2
3	[3, 4, 5, 6]	4
4	[4, 6]	2
5	[3, 4, 5, 6]	4
6	[4, 6]	2

Hamma xonalar orasida eng kichkina $p[i]$ qiymati 2ga teng, va bu $i \in \{1, 2, 4, 6\}$ lar uchun amalga oshirila olinadi. Shunday qilib, funksiya $[0, 1, 1, 0, 1, 0, 1]$ massivni javob sifatida qaytarishi kerak.

Misol 3

```
find_reachable([0, 0, 0], [0], [1], [0])
```

Quyidagi jadval har bir xonadan bora olinadigan xonalarni ko'rsatadi:

Boshlang'ich xona i	Bora olinadigan xonalar	$p[i]$
0	[0, 1]	2
1	[0, 1]	2
2	[2]	1

Hamma xonalar orasida eng kichkina $p[i]$ qiymati 1ga teng, va bu $i = 2$ uchun amalga oshirilgan. Shunday qilib, funksiya $[0, 0, 1]$ massiv qaytarishi kerak.

Chegaralar

- $2 \leq n \leq 300\,000$

- $1 \leq m \leq 300\,000$
- $0 \leq r[i] \leq n - 1$ har bir $0 \leq i \leq n - 1$ uchun
- $0 \leq u[j], v[j] \leq n - 1$ va $u[j] \neq v[j]$ har bir $0 \leq j \leq m - 1$ uchun
- $0 \leq c[j] \leq n - 1$ har bir $0 \leq j \leq m - 1$ uchun

Qism masala

1. (9 ball) $c[j] = 0$ har bir $0 \leq j \leq m - 1$ uchun, va $n, m \leq 200$
2. (11 ball) $n, m \leq 200$
3. (17 ball) $n, m \leq 2000$
4. (30 ball) $c[j] \leq 29$ (har bir $0 \leq j \leq m - 1$ uchun) va $r[i] \leq 29$ (har bir $0 \leq i \leq n - 1$ uchun)
5. (33 ball) Hech qanday qo'shimcha chegara yo'q.

Namunaviy greyder

Namunaviy greyder inputni quyidagi formatda o'qiydi:

- 1-qator: $n \ m$
- 2-qator: $r[0] \ r[1] \ \dots \ r[n - 1]$
- $3 + j$ - qatorlar ($0 \leq j \leq m - 1$): $u[j] \ v[j] \ c[j]$

Namunaviy greyder `find_reachable` funksiyasi qaytargan massivni quyidagi formatda chiqaradi:

- 1-qator: $sa[0] \ sa[1] \ \dots \ sa[n - 1]$