Sfinkso galvosūkis

Didysis Sfinksas tau paruošė galvosūkį. Tau duotas grafas su N viršūnių. Viršūnės sunumeruotos nuo 0 iki $N-1$. Grafe yra M briaunų, sunumeruotų nuo 0 iki $M-1$. Kiekviena briauna jungia skirtingų viršūnių porą ir yra dvikryptė. Kitaip tariant, kiekvienam j nuo 0 iki $M-1$ imtinai, briauna j jungia viršūnes $X[j]$ ir $Y[j]$. Kiekvieną viršūnių porą jungia ne daugiau kaip viena briauna. Dvi viršūnės vadinamos **gretimomis**, jeigu jos sujungtos briauna.

Viršūnių seka $v_0, v_1, \ldots, v_k$ (čia $k \geq 0$) vadinama **keliu**, jei sekoje greta esančios viršūnių poros v_l ir v_{l+1}, (visiems l, kur $0 \leq l < k$) yra gretimos. Sakome, kad kelias $v_0, v_1, \ldots, v_k$ **jungia** viršūnes v_0 ir v_k. Jums duotame grafe kiekvieną viršūnių porą jungia kažkoks kelias.

Yra $N+1$ spalvų, sunumeruotų nuo 0 iki N. Spalva N yra ypatinga ir mes ją vadiname **Sfinkso spalva**. Kiekviena viršūnė nuspalvinta viena iš spalvų. i-oji viršūnė ($0 \leq i < N$) nuspalvinta spalva $C[i]$. Skirtingos viršūnės gali būti tos pačios spalvos ir gali būti spalvų, kuriomis nenuspalvinta nei viena viršūnė. Jokia viršūnė nėra nuspalvinta Sfinkso spalva, taigi $0 \leq C[i] < N$ ($0 \leq i < N$).

Kelias $v_0, v_1, \ldots, v_k$ (kur $k \geq 0$) vadinamas **vienspalviu** jeigu visos jo viršūnės nuspalvintos ta pačia spalva, t. y. $C[v_l] = C[v_{l+1}]$ (kiekvienam l, kuriam $0 \leq l < k$). Taip pat sakome, kad viršūnės p ir q ($0 \leq p < N$, $0 \leq q < N$) priklauso tam pačiam **vienspalviam komponentui** tada ir tik tada, jei jas jungia vienspalvis kelias.

Jūs žinote viršūnes ir briaunas, bet nežinote, kokios spalvos yra kiekviena viršūnė. Jūs norite sužinoti viršūnių spalvas atlikdami **perspalvinimo eksperimentus**.

Perspalvinimo eksperimento metu jūs galite perspalvinti bet kiek viršūnių. Norint atlikti perspalvinimo eksperimentą, jūs pirmiausiai pasirenkate N dydžio masyvą E, kur kiekvienam i ($0 \leq i < N$), $E[i]$ yra tarp -1 ir N **imtinai**. Tada viršūnės i spalva tampa $S[i]$, kur $S[i]$ lygi:

- $C[i]$, taigi pradinei viršūnės i spalvai, jei $E[i] = -1$, arba
- $E[i]$ kitu atveju.

Atkreipkite dėmesį, kad tai reiškia, kad perspalvindami galite naudoti Sfinkso spalvą.

Galiausiai Didysis Sfinksas praneša vienspalvių komponentų kiekį grafe po to, kai kiekviena viršūnė i nuspalvinama $S[i]$ ($0 \leq i < N$) spalva. Kiekvienas spalvinimas naudojamas tik tam nuspalvinimo eksperimentui, taigi **pasibaigus eksperimentui visos viršūnės vėl yra pradinių spalvų**.

Nustatykite grafo viršūnių spalvas atlikdami daugiausiai $2\,750$ perspalvinimo eksperimentų. Jūs taip pat galite gauti dalinių taškų, jei kiekvienai gretimų viršūnių porai nustatote, ar jos turi tą pačią spalvą.

Realizacija

Jums reikia parašyti šią procedūrą.

```
std::vector<int> find_colours(int N,
    std::vector<int> X, std::vector<int> Y)
```

- N: viršūnių grafe skaičius.
- X, Y: M ilgių masyvai, nusakantys briaunas.
- Ši procedūra turi grąžinti N ilgio masyvą G, nurodantį grafo viršūnių spalvas.
- Ši procedūra kiekvienam testui iškviečiama lygiai vieną kartą.

Aukščiau esanti procedūra gali iškviesti šią procedūrą, kad atliktų perspalvinimo eksperimentus:

```
int perform_experiment(std::vector<int> E)
```

- E: N ilgio masyvas, nurodantis, kaip perspalvinti viršūnes.
- Ši procedūra grąžina vienspalvių komponentų skaičių po to, kai viršūnės perspalvinamos pagal masyve E nurodytas spalvas.
- Ši procedūra gali būti iškviesta daugiausiai $2\,750$ kartų.

Vertinimo programa **nėra adaptyvi**, taigi viršūnės yra nuspalvinamos prieš iškviečiant `find_colours`.

Ribojimai

- $2 \le N \le 250$
- $N - 1 \le M \le \frac{N \cdot (N-1)}{2}$
- $0 \le X[j] < Y[j] < N$ visiems j, kur $0 \le j < M$.
- $X[j] \neq X[k]$ arba $Y[j] \neq Y[k]$ visiems j ir k, kur $0 \le j < k < M$.
- Kiekvieną viršūnių porą jungia kažkoks kelias.
- $0 \le C[i] < N$ visiems i, kur $0 \le i < N$.

Dalinės užduotys

Dalinė užduotis	Taškai	Papildomi ribojimai
1	3	$N = 2$
2	7	$N \le 50$
3	33	Grafas yra kelias: $M = N - 1$ ir viršūnės j ir $j+1$ yra gretimos ($0 \le j < M$).
4	21	Grafas yra pilnas: $M = \frac{N \cdot (N-1)}{2}$ ir bet kurios dvi viršūnės yra gretimos.
5	36	Jokių papildomų ribojimų.

Kiekvienoje dalinėje užduotyje jūs galite surinkti dalį taškų, jei jūsų programa kiekvienai viršūnių porai teisingai nustato, ar tos viršūnės turi tą pačią spalvą.

Jūs gausite visus dalinės užduoties taškus, jei visuose jos testuose procedūros `find_colours` grąžintas masyvas G sutaps su masyvu C (t. y. $G[i] = C[i]$ visiems i, kur $0 \le i < N$). Kitu atveju gausite 50% dalinės užduoties taškų, jeigu visiems tos dalinės užduoties testams galioja:

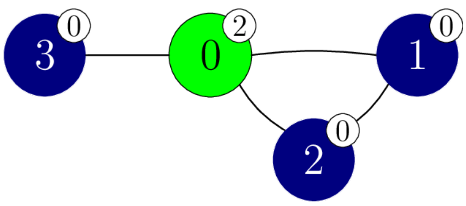
- $0 \le G[i] < N$ visiems i, kur $0 \le i < N$;
- Visiems j, kur $0 \le j < M$:
    - $G[X[j]] = G[Y[j]]$ tada ir tik tada, kai $C[X[j]] = C[Y[j]]$.

Pavyzdys

Panagrinėkime šį iškvietimą.

```
find_colours(4, [0, 1, 0, 0], [1, 2, 2, 3])
```

Tarkime, kad šiam pavyzdžiui (mums nežinomos) viršūnių spalvos yra $C = [2, 0, 0, 0]$. Šis scenarijus pavaizduotas žemiau esančioje diagramoje. Spalvas taip pat žymi prie kiekvienos viršūnės esantis skaičius baltame fone.



Procedūra gali iškviesti `perform_experiment` tokiu būdu.

```
perform_experiment([-1, -1, -1, -1])
```

Šiame iškvietime nei viena viršūnė nėra perspalvinama, taigi visos viršūnės lieka nuspalvintos pradinėmis spalvomis.

Panagrinėkime 1-ą ir 2-ą viršūnes. Jos abi nuspalvintos 0-ine spalva ir kelias $1, 2$ yra vienspalvis. Taigi viršūnės 1 ir 2 yra tame pačiame vienspalviame komponente.

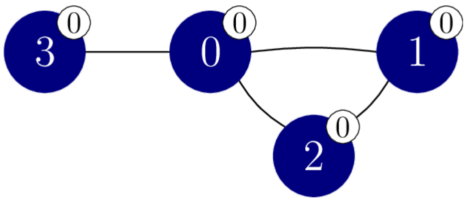
Panagrinėkime 1-ą ir 3-ią viršūnes. Nors abi viršūnės nuspalvintos 0-ine spalva, jos yra skirtinguose vienspalviuose komponentuose, nes nėra jokio vienspalvio kelio, kuris jas jungia.

Iš viso yra 3 vienspalviai komponentai, kuriuos sudaro viršūnės $\{0\}$, $\{1, 2\}$, bei $\{3\}$. Taigi šis iškvietimas grąžina 3.

Dabar procedūra gali iškviesti `perform_experiment` tokiu būdu.

```
perform_experiment([0, -1, -1, -1])
```

Šiame iškvietime tik 0-inė viršūnė perspalvinama 0-ine spalva, taigi mūsų grafas tampa nuspalvintas, kaip pavaizduota žemiau esančioje diagramoje.

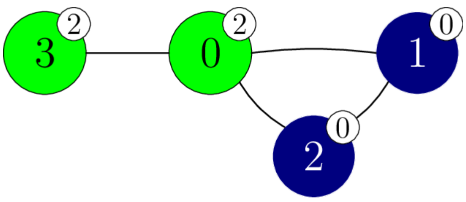


Šis iškvietimas grąžina 1, nes visos viršūnės priklauso tam pačiam vienspalviam komponentui. Mes dabar galime nustatyti, kad 1-a, 2-a ir 3-ia viršūnės yra nuspalvintos 0-ine spalva.

Procedūra tada gali iškviesti `perform_experiment` tokiu būdu.

```
perform_experiment([-1, -1, -1, 2])
```

Šiame iškvietime 3-ia viršūnė perspalvinama 2-aja spalva, taigi mūsų grafas tampa nuspalvintas, kaip pavaizduota žemiau esančioje diagramoje.



Šis iškvietimas grąžina 2, nes yra 2 vienspalviai komponentai, kuriems atitinkamai priklauso viršūnės $\{0, 3\}$ bei $\{1, 2\}$. Mes galime nustatyti, kad 0-inė viršūnė nuspalvinta 2-aja spalva.

Procedūra `find_colours` grąžina masyvą $[2, 0, 0, 0]$. Kadangi $C = [2, 0, 0, 0]$, surenkami visi taškai.

Taip pat yra daugiau nei vienas masyvas, kurį grąžinus galima gauti 50% taškų, pavyzdžiui $[1, 2, 2, 2]$ arba $[1, 2, 2, 3]$.

Pavyzdinė vertinimo programa

Pradinių duomenų formatas:

```
N  M
C[0]  C[1] ... C[N-1]
X[0]  Y[0]
X[1]  Y[1]
...
X[M-1]  Y[M-1]
```

Rezultatų formatas

```
L  Q
G[0]  G[1] ... G[L-1]
```

Čia L yra procedūros `find_colours` grąžinto masyvo G ilgis, o Q yra procedūros `find_colours` atliktų `perform_experiment` iškvietimų kiekis.