Sfingina uganka

Velika Sfinga ima zagonetko za vas. Podan vam je graf z N vozlišči. Vozlišča so oštevilčena od 0 do $N-1$. V grafu je M povezav, oštevilčenih od 0 do $M-1$. Vsaka povezava povezuje par različnih vozlišč in je dvosmerna. Natančneje, za vsak j od 0 do $M-1$ (vključno) povezava j povezuje vozlišči $X[j]$ in $Y[j]$. Vsak par vozlišč je povezan z največ eno povezavo. Dve vozlišči sta imenovani **sosednji**, če sta povezani s povezavo.

Zaporedje vozlišč $v_0, v_1, \ldots, v_k$ (za $k \geq 0$) imenujemo **pot**, če sta vsaki dve zaporedni vozlišči v_l in v_{l+1} sosednji (za vsak l velja $0 \leq l < k$). Pravimo, da pot $v_0, v_1, \ldots, v_k$ **povezuje** vozlišči v_0 in v_k. V podanem grafu je vsak par vozlišč povezan z neko potjo.

Obstaja $N+1$ barv, oštevilčenih od 0 do N. Barva N je posebna in se imenuje **Sfingina barva**. Vsakemu vozlišču je dodeljena ena barva. Natančneje, vozlišče i ($0 \leq i < N$) ima barvo $C[i]$. Več vozlišč ima lahko isto barvo, še vedno pa se lahko zgodi, da nekatere barve niso dodeljene kateremu koli vozlišču. Nobeno vozlišče nima Sfingine barve, t.j. $0 \leq C[i] < N$ ($0 \leq i < N$).

Pot $v_0, v_1, \ldots, v_k$ (za $k \geq 0$) je imenovana **monokromatska** če so vsa njena vozlišča iste barve, tj. $C[v_l] = C[v_{l+1}]$ (za vsak l velja $0 \leq l < k$). Poleg tega pravimo, da sta vozlišči p in q ($0 \leq p < N$, $0 \leq q < N$) v isti **monokromatski komponenti**, če in samo če sta povezani z monokromatsko potjo.

Poznate vozlišča in povezave, ne poznate pa barv vozliš. Želite ugotoviti barve vozlišč, z uporabo **Bojanovih eksperimentov**.

Pri izvajanju Bojanovega eksperimeta, lahko prebarvate poljubno število vozlišč. Natančneje, pri izvajanju Bojanovega eksperimeta najprej izberete polje E, velikosti N, kjer je za vsak i ($0 \leq i < N$) $E[i]$ med -1 in N **vključno**. Potem barva vsakega vozlišča i postane $S[i]$, kjer je vrednost $S[i]$:

- $C[i]$, t.j. prvotna barva i, če $E[i] = -1$,
- $E[i]$, sicer.

Upoštevajte, da pri svojem barvanju lahko uporabite Sfingino barvo.

Na koncu, po prebarvanju vseh vozlišč i na $S[i]$ ($0 \leq i < N$), Velika Sfinga pove število monokromatskih komponent grafa. Novo barvanje velja samo za ta določeni Bojanov eksperiment, kar pomeni: **po končanem eksperimentu se barve vseh vozlišč povrnejo na prvotno stanje**.

Vaša naloga je identificirati barve vozlišč grafa, z izvedbo največ $2,750$ Bojanovih eksperimentov. Prejmete lahko tudi delni rezultat, če za vsak par sosednjih vozlišč pravilno ugotovite, ali sta vozlišči enake barve.

Podrobnosti implementacije

Implementirati morate naslednjo funkcijo:

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

- N: število vozlišč grafa.
- X, Y: polji dolžine M, ki opisujeta povezave.
- Ta funkcija mora vrniti polje G, dolžine N, ki predstavlja barve vozlišč grafa.
- Ta funkcija se kliče natanko enkrat za vsak testni primer.

Zgornja funkcija lahko kliče naslednjo funkcijo, ki izvede Bojanov eksperiment:

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

- E: polje dolžine N, ki določa, kako naj bodo vozlišča prebarvana.
- Ta funkcija vrne število monokromatskih komponent, po prebarvanju vozlišč skladno z E.
- To funkcijo lahko pokličete največ $2,750$-krat.

Ocenjevalnik ni **prilagodljiv**, torej so barve vozlišč določene preden se izvede klic `find_colours`.

Omejitve

- $2 \leq N \leq 250$
- $N-1 \leq M \leq \frac{N \cdot (N-1)}{2}$
- $0 \leq X[j] < Y[j] < N$ kjer za vsak j velja $0 \leq j < M$.
- $X[j] \neq X[k]$ ali $Y[j] \neq Y[k]$ kjer za vsak j in k velja $0 \leq j < k < M$.
- Vsak par vozlišč je povezan z neko potjo.
- $0 \leq C[i] < N$ kjer za vsak i velja $0 \leq i < N$.

Podnaloge

Podnaloga	Točke	Dodatne omejitve
1	3	$N = 2$
2	7	$N \leq 50$
3	33	Graf je pot: $M = N - 1$ in vozlišči j in $j+1$ sta sosednji ($0 \leq j < M$).
4	21	Graf je poln: $M = \frac{N \cdot (N-1)}{2}$ in katerakoli dve vozlišči sta sosednji.
5	36	Brez dodatnih omejitev.

V vsaki podnalogi lahko pridobite delne točke, če vaša rešitev pravilno določi za vsak par sosednjih vozlišč, ali sta vozlišči enake barve.

Natančneje, vse točke podnaloge pridobite, če v vseh testnih primerih, polje G, ki ga vrne `find_colours`, popolnoma enako polju C (tj. $G[i] = C[i]$, kjer za vsak i velja $0 \leq i < N$). V nasprotnem primeru, dobite 50% točk za neko podnalogo če pri vseh njenih testnih primerih veljajo naslednji pogoji:

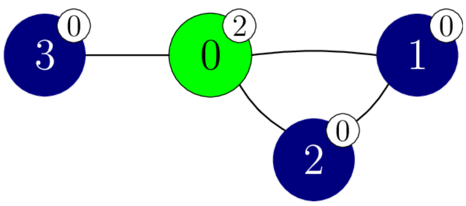
- $0 \leq G[i] < N$ za vsak i velja $0 \leq i < N$;
- Za vsak j velja $0 \leq j < M$:
    - $G[X[j]] = G[Y[j]]$ če in samo če $C[X[j]] = C[Y[j]]$.

Primer

Razmislite o naslednjem klicu.

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

Za ta primer predpostavimo, da so (skrite) barve vozlišč podane z $C = [2, 0, 0, 0]$. Upoštevajte naslednjo sliko.released/ioi24day2/hieroglyphs/pdf Barve so dodatno predstavljene s številkami v belih označbah vozliščih.



Funkcija lahko kliče `perform_experiment` na naslednji način:

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

V tem klicu nobeno vozlišče ni prebarvano, saj vsa vozlišča ohranijo svoje originalne barve.

Poglejmo vozlišče 1 in vozlišče 2. Obe vozlišči sta barve 0, in pot $1, 2$ je monokromatska pot. Kot rezultat, sta vozlišči 1 in 2 v isti monokromatski komponenti.

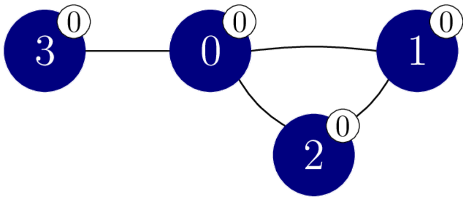
Poglejmo vozlišče 1 in vozlišče 3. Čeprav sta obe vozlišči barve 0, sta v različnih monokromatskih komponentah, saj ne obstaja monokromatska pot, ki bi ju povezovala.

Skupno obstajajo 3 monokromatske komponente, ki jih sestavljajo vozlišča $\{0\}$, $\{1, 2\}$ in $\{3\}$. Zatorej ta klic vrne 3.

Sedaj lahko funkcija kliče `perform_experiment` na naslednji način:

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

Pri tem klicu je vozlišče 0 prebarvano v barvo 0, kar ima za posledico barvanje, prikazano na naslednji sliki.

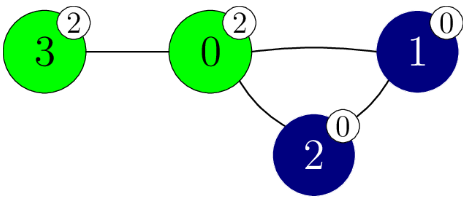


Ta klic vrne 1, saj so vsa vozlišča v isti monokromatski komponenti. Sedaj lahko ugotovimo, da so vozlišča 1, 2, in 3 barve 0.

Funkcija `perform_experiment` se nato lahko pokliče na naslednji način:

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

Pri tem klicu je vozlišče 3 prebarvano v barvo 2, kar vodi do barvanja, prikazanega na naslednji sliki.



Ta klic vrne 2, ker obstajata 2 monokromatsi komponenti, sestavljeni iz vozlišč $\{0, 3\}$ in $\{1, 2\}$. Lahko zaključimo, da ima vozlišče 0 barvo 2.

Funkcija `find_colours` nato vrne polje $[2, 0, 0, 0]$. Ker je $C = [2, 0, 0, 0]$, se dodeli vse točke.

Upoštevajte, da obstaja več možnih odgovorov, za katere bi bilo dodeljenih 50% točk, na primer $[1, 2, 2, 2]$ in $[1, 2, 2, 3]$.

Vzorčni ocenjevalec

Oblika vhoda:

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

Oblika izhoda:

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

Tukaj je L dolžina polja G, ki ga vrne `find_colours`, in Q je število klicev `perform_experiment`.