

Sphinx's Riddle

Ұлы Сфинкстің сізге арналған жұмбақтары бар. Сізге N төбелері бар граф берілген. Төбелер 0 -ден бастап $N - 1$ -ге дейін нөмірленген. Графта 0 -ден бастап $M - 1$ -ге дейін нөмірленген M қырлар бар. Әрбір қыр әртүрлі төбелер жұбын байланыстырады және екі бағытты болады. Атап айтқанда, 0 -ден бастап $M - 1$ -ге дейінгі (қоса алғанда) әрбір j үшін j -ші қыр $X[j]$ және $Y[j]$ төбелерін қосады. Кез келген төбелер жұбын байланыстыратын ең көбі бір қыр бар. Екі төбе **көршілес** деп аталады, егер олар қыр арқылы байланысқан болса.

v_0, v_1, \dots, v_k ($k \geq 0$ үшін) төбелер тізбегі **жол** деп аталады, егер әрбір екі қатар келетін v_l және v_{l+1} ($0 \leq l < k$ болатын әрбір l үшін) төбелері көршілес болса. v_0, v_1, \dots, v_k жолы v_0 және v_k төбелерін **байланыстырады** деп айтамыз. Сізге берілген графта төбелердің әрбір жұбы қандай да бір жол арқылы байланысқан.

0 -ден бастап N -ге дейін нөмірленген $N + 1$ түстер бар. N -ші түс ерекше және **Сфинкс түсі** деп аталады. Әрбір төбеге түс тағайындалған. Атап айтқанда, i -ші ($0 \leq i < N$) төбенің түсі $C[i]$. Бірнеше төбелердің түстері бірдей болуы мүмкін және ешбір төбеге тағайындалмаған түстер болуы мүмкін. Ешбір төбеде сфинкстің түсі жоқ, яғни $0 \leq C[i] < N$ ($0 \leq i < N$).

v_0, v_1, \dots, v_k ($k \geq 0$ үшін) жолы **монохроматикалық** деп аталады, егер оның барлық төбелерінің түстері бірдей болса, яғни $C[v_l] = C[v_{l+1}]$ ($0 \leq l < k$ болатын әрбір l үшін). Сонымен қатар, p және q төбелері ($0 \leq p < N$, $0 \leq q < N$) бір **монохроматикалық компонентте** болады деп айтамыз, егер олар монохроматикалық жолмен байланысқан болса ғана.

Сіз төбелер мен қырларды білесіз, бірақ ешқандай төбенің түсін білмейсіз. Сіз **бояу эксперименттерін** орындау арқылы төбелердің түстерін білгіңіз келеді.

Бояу экспериментінде көптеген төбелерді еркін бояуға болады. Атап айтқанда, бояуды экспериментін орындау үшін алдымен ұзындығы N болатын E массивін таңдайсыз, мұнда әрбір i ($0 \leq i < N$) үшін $E[i]$ мәні -1 және N арасында (**қоса алғанда**) жатады. Содан кейін әрбір i -ші төбенің түсі $S[i]$ болады, мұнда $S[i]$ мәні:

- $C[i]$, егер $E[i] = -1$ болса, яғни i -ші төбенің бастапқы түсі.
- $E[i]$, басқа жағдайда.

Бояу кезінде Сфинкс түсін пайдалануға болатындығын ескеріңіз.

Соңында, Ұлы Сфинкс әрбір i -ші төбенің түсі $S[i]$ ($0 \leq i < N$) етіп орнатқаннан кейін графтағы монохроматикалық компоненттердің санын хабарлайды. Жаңа бояу тек осы нақты бояу экспериментінде ғана қолданылады, сондықтан **барлық төбелердің түстері эксперимент аяқталғаннан кейін бастапқы түстеріне оралады.**

Сіздің тапсырмаңыз - ең көбі 2 750 бояу эксперименттерін орындау арқылы графтағы төбелердің түстерін анықтау. Әрбір көршілес төбелер жұбы үшін олардың түстері бірдей екендігін дұрыс анықтасаңыз, жартылай ұпай алуға болады.

Implementation Details

Сізге келесі функцияны іске асыру керек.

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

- N : графтағы төбелер саны.
- X, Y : қырларды сипаттайтын ұзындықтары M болатын массивтер.
- Бұл функция төбелердің түстерін сипаттайтын ұзындығы N болатын G массивін қайтаруы керек.
- Бұл функция әрбір тест үшін бір рет шақырылады

Жоғарыдағы функция бояу эксперименттерін орындау үшін келесі функцияны шақыра алады:

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

- E : төбелерді қалай бояу керектігін көрсететін ұзындығы N болатын массив.
- Бұл функция төбелерді E -ге бояғаннан кейін монохроматикалық компоненттер санын қайтарады.
- Бұл функцияны ең көбі 2 750 рет шақыруға болады.

Грейдер **бейімделмейді**, яғни төбелердің түстері `find_colours` шақыру жасалмас бұрын бекітіледі.

Constraints

- $2 \leq N \leq 250$
- $N - 1 \leq M \leq \frac{N \cdot (N - 1)}{2}$
- $0 \leq X[j] < Y[j] < N$, $0 \leq j < M$ орындалатын әрбір j үшін.
- $X[j] \neq X[k]$ or $Y[j] \neq Y[k]$, $0 \leq j < k < M$ орындалатын әрбір j және k үшін.
- Төбелердің әрбір жұбы қандай да бір жол арқылы байланысқан.
- $0 \leq C[i] < N$, $0 \leq i < N$ орындалатын әрбір i үшін.

Subtasks

Ішкі есеп	Ұпай	Қосымша шектеулер
1	3	$N = 2$
2	7	$N \leq 50$
3	33	Граф жол болады: $M = N - 1$ және j мен $j + 1$ төбелері көршілес ($0 \leq j < M$).
4	21	Граф толық болады: $M = \frac{N \cdot (N-1)}{2}$ және әрбір екі төбе көршілес.
5	36	Қосымша шектеулер жоқ

Әрбір ішкі есепте, егер сіздің бағдарламаңыз көршілес төбелердың әрбір жұбы үшін олардың түсі бірдей екенін дұрыс анықтаса, жартылай ұпай алуға болады.

Дәлірек айтқанда, егер оның барлық тесттеріне `find_colours` арқылы қайтарылған G массиві C массивімен бірдей болса (яғни $0 \leq i < N$ болатын барлық i үшін $G[i] = C[i]$) сіз сол ішкі есептің толық ұпайын аласыз. Әйтпесе, барлық тесттерде келесі шарттар орындалса, ішке есеп үшін 50% ұпай аласыз:

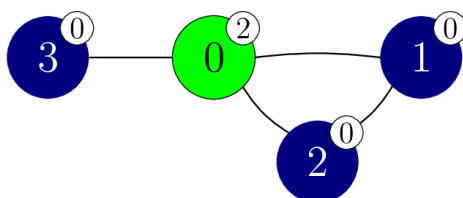
- $0 \leq G[i] < N$, $0 \leq i < N$ болатын әрбір i үшін;
- $0 \leq j < M$ болатын әрбір j үшін:
 - $G[X[j]] = G[Y[j]]$, егер және тек $C[X[j]] = C[Y[j]]$ болған жағдайда ғана.

Example

Келесі мысалды қарастырыңыз.

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

Бұл мысал үшін төбелердің (жасырын) түстері $C = [2, 0, 0, 0]$ деп есептейік. Бұл сценарий келесі суретте көрсетілген. Түстер әр төбеге бекітілген қосымша ақ жапсырмалардағы сандармен көрсетіледі.



`perform_experiment` функциясы келесідей шақырылуы мүмкін.

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

Бұл шақыруда ешбір төбенің түсі өзгертілмейді, өйткені барлық төбелер өздерінің бастапқы түстерін сақтайды.

1 төбесін және 2 төбесін қарастырыңыз. Олардың екеуінің де түсі 0 және 1,2 жолы монохроматикалық жол болады. Нәтижесінде 1 және 2 төбелері бір монохроматикалық компонентте болады.

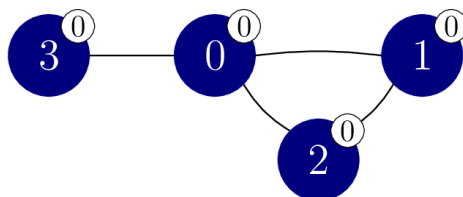
1 төбесін және 3 төбесін қарастырыңыз. Олардың екеуінің де түсі 0 болса да, олар әртүрлі монохроматикалық компонентіде жатады, өйткені оларды байланыстыратын монохроматикалық жол жоқ.

Жалпы 3 монохроматикалық компоненттер бар, олардың төбелері $\{0\}$, $\{1,2\}$ және $\{3\}$. Осылайша, бұл шақыру 3 қайтарады.

Енді `perform_experiment` функциясы келесідей шақырылуы мүмкін.

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

Бұл шақыруда тек 0 төбесі 0 түсіне боялады, бұл келесі суретте көрсетілген бояуға әкеледі.

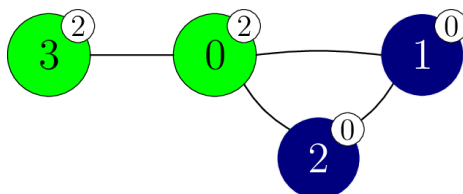


Бұл шақыру 1 қайтарады, өйткені барлық төбелер бір монохроматикалық компонентіде жатады. Енді 1, 2 және 3 төбелерінің түстері 0 екенін анықтай аламыз.

Енді `perform_experiment` функциясы келесідей шақырылуы мүмкін.

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

Бұл шақыруда 3 төбесі 2 түсіне боялады, бұл келесі суретте көрсетілген бояуға әкеледі.



Бұл шақыру 2 қайтарады, себебі 2 монохроматикалық компоненттер бар, олардың төбелері сәйкесінше $\{0,3\}$ және $\{1,2\}$. 0 төбесінің түсі 2 екенін анықтай аламыз.

`find_colours` функциясы $[2, 0, 0, 0]$ массивін қайтарады. $C = [2, 0, 0, 0]$ болғандықтан, толық ұпай беріледі.

Сондай-ақ 50% ұпай берілетін бірнеше қайтару мәндері бар екенін ескеріңіз, мысалы $[1, 2, 2, 2]$ немесе $[1, 2, 2, 3]$.

Sample Grader

Енгізу форматы:

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

Шығару форматы:

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

Мұнда L `find_colours` арқылы қайтарылған G массивінің ұзындығы және Q - `perform_experiment` шақырулар саны.