

În ul# Angry Cows

Problem name	Angry Cows
Input file	standard input
Output file	standard output
Time limit	6 seconds
Memory limit	256 megabytes

În ultimii ani s-a atestat o răspândire rapidă a bolii Extremely Green Oxen Illness (EGOI), care face vacile periculoase pentru drumeți. După mai multe incidente, s-a decis că trebuie separate zonele în care pasc vacile de partea Alpilor în care oamenii vor să călătorească.

Ți se dă o hartă a Alpilor. Sunt  $n$  zone pe hartă. Fiecare dintre ele poate fi o zonă populată de vaci, o zonă de călătorii, sau o zonă neutilizată. unele perechi de zone sunt conectate prin trasee bidirecționale. fiecare traseu are o lungime non-negativă. (în termeni de grafuri, o hartă este un graf neorientat cu muchii ponderate.)

Odată ce construieți un zid într-o zonă, zona devine inaccesibilă excursioniștilor și vacilor - aceștia nu vor mai putea să meargă printr-o astfel de zonă.

Sarcina ta este să selectezi setul de zone în care vor fi construite zidurile. Acest set de zone urmează să satisfacă următoarele condiții:

- să fie format doar din zone neutilizate.
- trebuie să separe zonele populate de vaci de zonele pentru călătorii. Astfel, vacile nu vor mai avea posibilitatea să se plimbe pe drumuri din zonele populate de ele în zonele pentru călătorii (fără a rece prin o zonă cu ziduri).
- Setul trebuie să nu separe nici o zonă pentru călătorii de alte zone de același fel. Astfel, călătorii vor putea merge în continuare de-a lungul drumului dintr-o zonă de călătorii în alta (fără a trece prin o zonă cu ziduri).

Dacă există mai multe modalități de a atinge obiectivul de mai sus, ne va păsa de ușurința întreținerii pereților. Zidurile vor fi întreținute de echipaje specializate. Există un astfel de echipaj cu sediul în fiecare zonă pentru călătorii.

Pentru fiecare zonă  $A$  vom defini depărtarea ei ca lungimea minimală a căii între  $A$  și o zonă pentru călătorii. (Lungimea unei căi este suma lungimilor drumurilor sale. Rețineți că aceste căi **pot** traversa pereții și zonele cu vaci - echipajul de întreținere a zidurilor are toate abilitățile și echipamentele necesare pentru a face acest lucru.)

Depărtarea unui set de zone este astfel **valoarea maximală** a depărtării după toate zonele din set.

Dintre toate seturile de zone cu ziduri care au proprietățile necesare, găsiți și returnați unul cu **cea mai mică distanță** posibilă. Dacă există multe astfel de seturi de zone, puteți returna oricare dintre ele

De notat că numărul de zone nu contează. În particular, **nu** este obligatoriu să folosiți cât mai puține ziduri posibil.

## Input

Prima linie din input conține două numere întregi  $n$  și  $m$  ( $2 \leq n \leq 3 \cdot 10^5$ ,  $n - 1 \leq m \leq 3 \cdot 10^5$ ) - numărul de zone și de drumuri, respectiv. Zonele sunt numerotate de la 1 la  $n$ .

Cea de a doua linie conține  $n$  numere întregi, separate prin spațiu  $t_1, \dots, t_n$ , unde  $t_i$  este  $-1$  dacă a  $i$ -a zonă este populată de vaci,  $0$  dacă este neutilizată, și  $1$  dacă este o zonă pentru călătorii.

Liniile rămase,  $m$  la număr descriu drumurile. A  $j$ -a dintre ele conține trei numere separate prin spațiu  $a_j, b_j, \ell_j$  ( $1 \leq a_j < b_j \leq n$ ,  $0 \leq \ell_j \leq 10^9$ ), marcând un drum între zona  $a_j$  și  $b_j$  de lungime  $\ell_j$ .

Se garantează că:

- între oricare două zone există cel mult un drum,
- inițial este posibil să mergi între oricare două zone folosind zero sau mai multe drumuri
- există cel puțin o zonă cu vaci,
- există cel puțin o zonă de călătorii

## Output

Dacă este imposibil să construiești zidurile conform cerințelor, afișați  $-1$ .

În caz contrar, prima linie din output trebuie să conțină un număr întreg  $k$  - numărul de ziduri pe care dorești să le construiești. Cea de a doua linie trebuie să conțină  $k$  numere întregi - numărul de zone în care se vor construi ziduri. (aceste numere trebuie să fi numere distincte între  $1$  și  $n$ , inclusiv. Nu este necesară o ordine specială de afișare.)

Output-ul va fi acceptat dacă este un set de ziduri cu o depărtare minimală.

## Scoring

Subtask 1 (7 puncte):  $n \leq 10$ .

Subtask 2 (22 puncte): toate lungimile sunt  $\ell_j = 0$ .

Subtask 3 (16 puncte): există o singură zonă pentru călătorii.

Subtask 4 (11 puncte): sunt exact  $n - 1$  drumuri (în termeni de grafuri - avem un arbore).

Subtask 5 (8 puncte): avem  $n, m \leq 2000$  și toate lungimile  $\ell_j = 1$ .

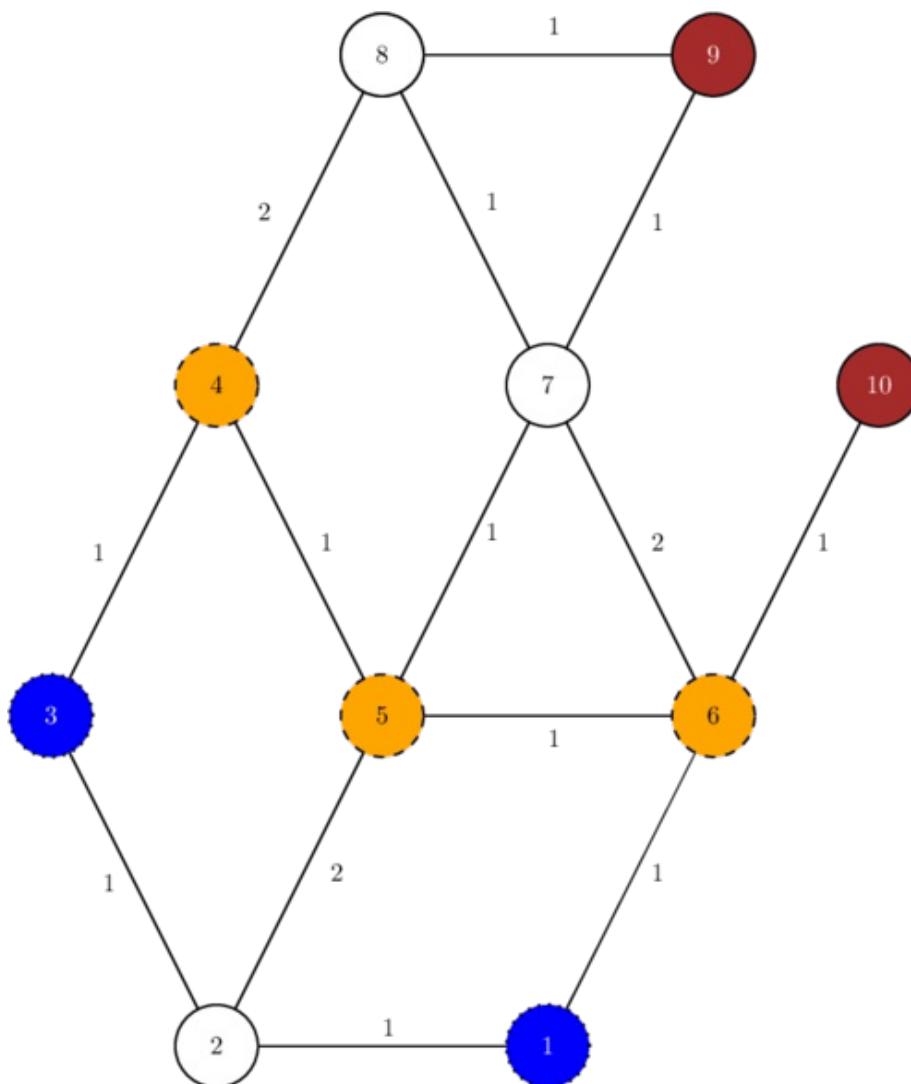
Subtask 6 (36 puncte): fără restricții adiționale.

## Exemplu

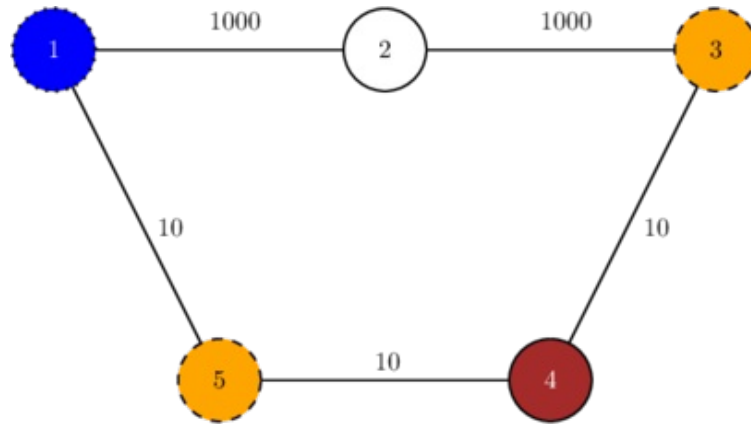
standard input	standard output
10 14 1 0 1 0 0 0 0 0 -1 -1 1 2 1 1 6 1 2 3 1 2 5 2 3 4 1 4 5 1 4 8 2 5 6 1 5 7 1 6 7 2 6 10 1 7 8 1 7 9 1 8 9 1	3 4 5 6
5 5 1 0 0 -1 0 1 2 1000 2 3 1000 3 4 10 4 5 10 1 5 10	2 3 5
4 3 1 0 -1 1 1 2 0 2 3 21 2 4 13	-1

## Note

ÎN toate figurile, algastru (cu puncte) e folosit pentru zonele de călătorii, cafenu (cu linii continui) pentru zonele populate de vaci și orange (cu linii întrerupte) pentru ziduri.



În primul exemplu, cea mai mică depărtare este 2, atinsă prin plasarea zidurilor în zonele 4, 5 și 6. De notat că nu pot fi plasate ziduri în zonele 4, 2 și 6, chiar dacă obținem o depărtare 1, deoarece va deveni imposibil de călătorit între zonele 1 și 3 fără a trece de vre-un zid..



În al doilea exemplu, depărtarea zonei 2 este 1000, și depărtarea zonei 3 este 30, și poate fi atinsă folosind calea 1-5-4-3. (Amintim că echipajele de întreținere pot trece prin ziduri și zone populate de vaci.). Prin urmare, ar trebui să așezăm ziduri în zonele 5 și 3 (nu 2), și depărtarea va fi 30.