Zapiralni časi

Madžarska je država z N mesti, oštevilčenimi od 0 do $N-1$. Mesta so povezana z $N-1$ *dvosmernimi* cestami, oštevilčenimi od 0 do $N-2$. Za vsak j, za katerega velja $0 \le j \le N-2$, cesta j povezuje mesto $U[j]$ in mesto $V[j]$ ter ima dolžino $W[j]$, kar pomeni, da omogoča potovanje med mesti v $W[j]$ enotah časa. Vsaka cesta povezuje dve različni mesti in vsak par mest je povezan z največ eno cesto.

Pot med dvema različnima mestoma a in b je zaporedje $p_0, p_1, \ldots, p_t$ različnih mest, pri čemer velja:

- $p_0 = a$,
- $p_t = b$,
- za vsak i ($0 \le i < t$) obstaja cesta, ki povezuje mesti p_i in p_{i+1}.

Možno je potovati iz katerega koli mesta v katero koli drugo mesto s pomočjo cest, tj. obstaja pot med vsakima dvema različnima mestoma. Za vsak par različnih mest lahko dokažemo, da je ta pot edinstvena.

Dolžina poti $p_0, p_1, \ldots, p_t$ je vsota dolžin t cest, ki povezujejo zaporedna mesta na poti.

Na Madžarskem mnogi ljudje potujejo na praznovanja ustanovitvenega dneva v dveh prazničnih mestih. Ko so praznovanja končana, se vrnejo domov. Vlada želi preprečiti motnje lokalnega prebivalstva s strani množice ljudi in načrtuje zaprtje vseh mest ob določenih urah. Vsako mesto bo vlada opremila z ne-negativnim **zapiralnim časom**. Vlada je odločila, da vsota vseh časov zaprtja ne sme biti večja od K. Natančneje, za vsak i med 0 in vključno N - 1 je čas zaprtja dodeljen mestu i ne-negativno celo število c[i]. Vsota vseh c[i] ne sme biti večja od K.

Naj bo a neko mesto in naj bo dana neka dodelitev zapiralnih časov. Rečemo, da je mesto b **dosegljivo** iz mesta a, če velja bodisi $b = a$, bodisi pot $p_0, \ldots, p_t$ med tema dvema mestoma (torej velja $p_0 = a$ in $p_t = b$) zadošča naslednjim pogojem:

- dolžina poti p_0, p_1 je največ $c[p_1]$,
- dolžina poti p_0, p_1, p_2 je največ $c[p_2]$,
- $\ldots$
- dolžina poti $p_0, p_1, p_2, \ldots, p_t$ je največ $c[p_t]$.

Letos sta glavni praznični prizorišči v mestih X in Y. Za vsako dodelitev zapiralnih časov je **ocena priročnosti** definirana kot vsota naslednjih dveh števil:

- Število mest, ki so dosegljiva iz mesta X.
- Število mest, ki so dosegljiva iz mesta Y.

Opomba: Če je mesto dosegljivo iz mesta X in hkrati iz mesta Y, šteje *dvakrat*.

Vaša naloga je izračunati največjo oceno priročnosti, ki jo lahko dosežemo z neko dodelitvijo časov zaprtja.

Podrobnosti implementacije

Implementirajte naslednjo funkcijo:

```
int max_score(int N, int X, int Y, int64 K, int[] U, int[] V, int[] W)
```

- N: število mest.
- X, Y: mesta z glavnimi prazničnimi prizorišči.
- K: zgornja meja vsote časov zaprtja.
- U, V: tabela dolžine $N-1$, ki opisuje povezave cest.
- W: tabela dolžine $N-1$, ki opisuje dolžine cest.
- Funkcija naj vrne največjo oceno priročnosti, ki jo lahko dosežemo z neko dodelitvijo časov zaprtja.
- Funkcija se lahko kliče **večkrat** v vsakem testnem primeru.

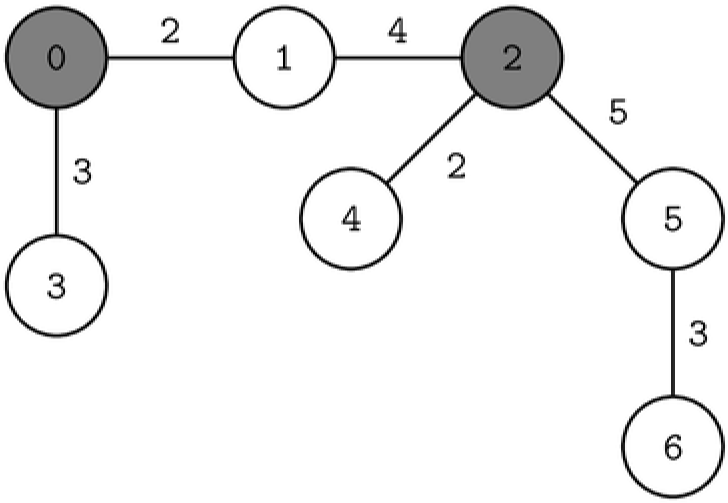
Na primer:

Primer

Upoštevajte naslednji klic:

```
max_score(7, 0, 2, 10,
          [0, 0, 1, 2, 2, 5], [1, 3, 2, 4, 5, 6], [2, 3, 4, 2, 5, 3])
```

To ustreza naslednjemu cestnemu omrežju:



Predpostavimo, da so zapiralni časi razporejeni na naslednji način:

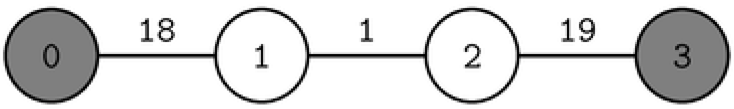
Mesto	0	1	2	3	4	5	6
Zapiralni čas	0	4	0	3	2	0	0

Opomba: Vsota vseh zapiralnih časov je enaka 9, kar ni več kot $K = 10$. Mesta 0, 1 in 3 so dosegljiva iz mesta X ($X = 0$), medtem ko so mesta 1, 2 in 4 dosegljiva iz mesta Y ($Y = 2$). Ocena priročnosti je torej enaka $3 + 3 = 6$. Ni dodelitve zapiralnih časov, kjer bi bila ocena priročnosti večja od 6, zato mora funkcija vrniti vrednost `6`.

Upoštevajte tudi naslednji klic:

```
max_score(4, 0, 3, 20, [0, 1, 2], [1, 2, 3], [18, 1, 19])
```

To ustreza naslednjemu cestnemu omrežju:



Predpostavimo, da so zapiralni časi razporejeni na naslednji način:

Mesto	0	1	2	3
Zapiralni čas	0	1	19	0

Mesto 0 je dosegljivo iz mesta X ($X = 0$), medtem ko sta mesti 2 in 3 dosegljivi iz mesta Y ($Y = 3$). Zato je ocena priročnosti enaka $1 + 2 = 3$. Ni dodelitve zapiralnih časov, kjer bi bila ocena priročnosti večja od 3, zato mora postopek vrniti 3.

Omejitve

- $2 \le N \le 200\,000$
- $0 \le X < Y < N$
- $0 \le K \le 10^{18}$
- $0 \le U[j] < V[j] < N$ (za vsak j, kjer $0 \le j \le N-2$)
- $1 \le W[j] \le 10^6$ (za vsak j, kjer $0 \le j \le N-2$)
- S cestami je mogoče potovati iz katerega koli mesta v katero koli drugo mesto.
- $S_N \le 200\,000$, kjer je S_N vsota N vseh klicev `max_score` v vsakem testnem primeru.

Podnaloge

Rečemo, da je cestno omrežje **linearno**, če cesta i povezuje mesti i in $i+1$ (za vsak i, tako da $0 \le i \le N-2$).

1. (8 točk) Dolžina poti od mesta X do mesta Y je večja od $2K$.
2. (9 točk) $S_N \le 50$, cestno omrežje je linearno.
3. (12 točk) $S_N \le 500$, cestno omrežje je linearno.
4. (14 točk) $S_N \le 3\,000$, cestno omrežje je linearno.
5. (9 točk) $S_N \le 20$
6. (11 točk) $S_N \le 100$
7. (10 točk) $S_N \le 500$
8. (10 točk) $S_N \le 3\,000$
9. (17 točk) Brez dodatnih omejitev.

Vzorčni ocenjevalnik

Naj bo C število scenarijev, tj. število klicev funkcije `max_score`. Vzorčni ocenjevalnik bere vhod naslednje oblike:

- vrstica 1: C

Opisi C scenarijev sledijo.

Vzorčni ocenjevalnik prebere opis vsakega scenarija naslednje oblike:

- vrstica 1: $N\ X\ Y\ K$
- vrstica $2+j$ ($0 \le j \le N-2$): $U[j]\ V[j]\ W[j]$

Vzorčni ocenjevalnik za vsak scenarij izpiše eno vrstico naslednje oblike:

- vrstica 1: vrednost, ki jo vrne funkcija `max_score`