Slēgšanas laiks

Ungārija ir valsts ar N pilsētām, kas numurētas no 0 līdz $N-1$.

Pilsētas ir savienotas ar $N-1$ *divvirzienu* ceļiem, kas numurēti no 0 līdz $N-2$. Visiem j, kuriem $0 \le j \le N-2$, ceļš j savieno pilsētu $U[j]$ un pilsētu $V[j]$ un ir ar garumu $W[j]$, tas ir, tas atļauj pārvietoties starp pilsētām $W[j]$ laika vienībās. Katrs ceļš savieno divas dažādas pilsētas, un katrs divu pilsētu pāris ir savienots ar ne vairāk kā vienu ceļu.

Maršruts starp divām dažādām pilsētām a un b ir virkne $p_0, p_1, \ldots, p_t$ ar atšķirīgām pilsētām tāda, ka:

- $p_0 = a$,
- $p_t = b$,
- visiem i ($0 \le i < t$), eksistē ceļš, kas savieno pilsētas p_i un p_{i+1}.

Ir iespējams pārvietoties no katras pilsētas uz jebkuru citu pilsētu izmantojot ceļus, tas ir, eksistē maršruts starp jebkurām divām atšķirīgām pilsētām. Var pierādīt, ka šis maršruts ir unikāls katram pārim ar atšķirīgām pilsētām.

Garums maršrutam $p_0, p_1, \ldots, p_t$ ir garumu summa t ceļiem, kas savieno secīgas pilsētas maršrutā.

Ungārijā daudzi cilvēki pārvietojas, lai apmeklētu Dibināšanas Dienas svētkus divās lielākajās pilsētās. Kad svinības ir beigušās, viņi atgriežas savās mājās. Valdība vēlas novērst cilvēku pūļus no vietējo iedzīvotāju traucēšanas, tādēļ viņi plāno slēgt visas pilsētas noteiktos laikos. Katrai pilsētai valdība piešķirs nenegatīvu **slēgšanas laiku**. Valdība ir nolēmusi, ka visu pilsētu slēgšanas laiku summa nedrīkst pārsniegt K. Precīzāk, katram i starp 0 un $N-1$, ieskaitot, slēgšanas laiks pilsētai i ir nenegatīvs vesels skaitlis $c[i]$. Visu $c[i]$ summa nedrīkst būt lielāka, kā K.

Apskatīsim pilsētu a un kaut kādu slēgšanas laiku konfigurāciju. Teiksim, ka pilsēta b ir **sasniedzama** no pilsētas a tad un tikai tad, ja vai nu $b = a$, vai maršruts $p_0, \ldots, p_t$ starp šīm divām pilsētām ($p_0 = a$ un $p_t = b$) izpilda šādus nosacījumus:

- garums maršrutam p_0, p_1 ir ne vairāk kā $c[p_1]$, un
- garums maršrutam p_0, p_1, p_2 ir ne vairāk kā $c[p_2]$, un
- $\ldots$
- garums maršrutam $p_0, p_1, p_2, \ldots, p_t$ ir ne vairāk kā $c[p_t]$.

Šogad galvenās divas svētku vietas atrodas pilsētā X un pilsētā Y. Katrai slēgšanas laiku konfigurācijai **komforta vērtību** definē kā šādu divu skaitļu summu:

- Pilsētu skaits, ko var sasniegt no pilsētas X.
- Pilsētu skaits, ko var sasniegt no pilsētas Y.

Ievērojiet to, ka, ja pilsēta ir sasniedzama no pilsētas X un ir sasniedzama no pilsētas Y, tad tā *divreiz* tiek ieskaitīta komforta vērtībā.

Jūsu uzdevums ir aprēķināt lielāko komforta vērtību, kāda var tikt sasniegta ar kaut kādu slēgšanas laiku konfigurāciju.

Implementācijas detaļas

Jums jāimplementē šāda procedūra:

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

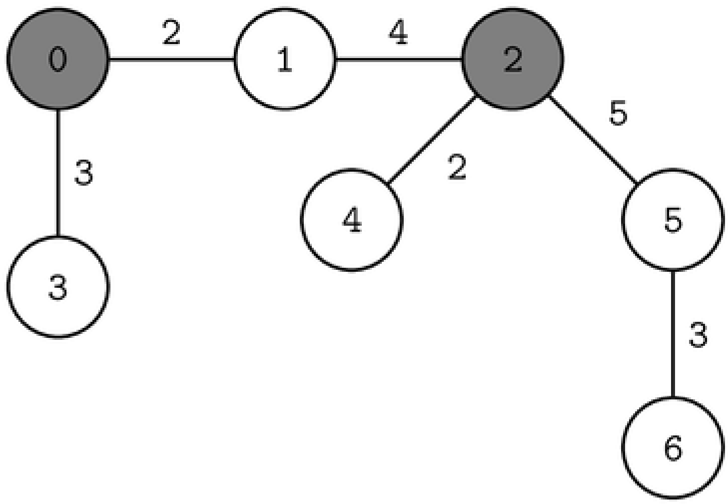
- N: pilsētu skaits.
- X, Y: pilsētas ar galvenajām svētku vietām.
- K: augšējā robeža slēgšanas laiku summai.
- U, V: masīvi garumā $N-1$, kas apraksta ceļu savienojumus.
- W: masīvs garumā $N-1$, kas apraksta ceļu garumus.
- Procedūrai jāatgriež maksimālā komforta vērtība, ko var sasniegt ar kaut kādu slēgšanas laiku konfigurāciju.
- Šī procedūra var tikt izsaukta **vairākas reizes** katrā testā.

Piemērs

Apskatīsim šādu procedūras izsaukumu:

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

Šis atbilst šādam ceļu tīklam:



Pieņemsim, ka slēgšanas laiki ir izvēlēti šādi:

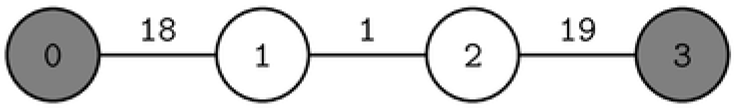
Pilsēta	0	1	2	3	4	5	6
Slēgšanas laiks	0	4	0	3	2	0	0

Ievērojiet to, ka slēgšanas laiku summa ir 9, kas nav vairāk kā $K = 10$. Pilsētas 0, 1 un 3 ir sasniedzamas no pilsētas X ($X = 0$), kamēr pilsētas 1, 2, un 4 ir sasniedzamas no pilsētas Y ($Y = 2$). Tāpēc komforta vērtība ir $3 + 3 = 6$. Neeksistē tāda slēgšanas laiku konfigurācija ar komforta vērtību, kas ir lielāka par 6, tādēļ procedūrai jāatgriež 6.

Apsveriet arī šādu procedūras izsaukumu:

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

Šis atbilst šādam ceļu tīklam:



Pieņemsim, ka slēgšanas laiki ir izvēlēti šādi:

Pilsēta	0	1	2	3
Slēgšanas laiks	0	1	19	0

Pilsēta 0 ir sasniedzama no pilsētas X ($X = 0$), kamēr pilsētas 2 un 3 ir sasniedzamas no pilsētas Y ($Y = 3$). Tāpēt komforta vērtība ir $1 + 2 = 3$. Neeksistē tāda slēgšanas laiku konfigurācija ar komforta vērtību vairāk kā 3, tādēļ procedūrai jāatgriež 3.

Ierobežojumi

- $2 \leq N \leq 200\,000$
- $0 \leq X < Y < N$
- $0 \leq K \leq 10^{18}$
- $0 \leq U[j] < V[j] < N$ (visiem j, kuriem $0 \leq j \leq N-2$)
- $1 \leq W[j] \leq 10^6$ (visiem j, kuriem $0 \leq j \leq N-2$)
- Ir iespējams pārvietoties no jebkuras pilsētas uz jebkuru citu pilsētu izmantojot ceļus.
- $S_N \leq 200\,000$, kur S_N ir visu N vērtību summa no procedūras `max_score` izsaukumiem katrā testā.

Apakšuzdevumi

Teiksim, ka ceļu tīkls ir **lineārs**, ja ceļš i savieno pilsētas i un $i+1$ (visiem i, kuriem $0 \leq i \leq N-2$).

1. (8 punkti) Garums maršrutam no pilsētas X uz pilsētu Y ir garāks par $2K$.
2. (9 punkti) $S_N \leq 50$, ceļu tīkls ir lineārs.
3. (12 punkti) $S_N \leq 500$, ceļu tīkls ir lineārs.
4. (14 punkti) $S_N \leq 3\,000$, ceļu tīkls ir lineārs.
5. (9 punkti) $S_N \leq 20$
6. (11 punkti) $S_N \leq 100$
7. (10 punkti) $S_N \leq 500$
8. (10 punkti) $S_N \leq 3\,000$
9. (17 punkti) Bez papildus ierobežojumiem.

Piemēra vērtētājprogramma

Ar C apzīmēsim scenāriju skaitu, tas ir, procedūras `max_score` izsaukumu skaitu. Piemēra vērtētājprogramma ielasīs ievaddatus šādā formātā:

- 1. rinda: C

Tālāk seko C scenāriju apraksts.

Piemēra vērtētājprogramma ielasa katra scenārija aprakstu šādā formātā:

- 1. rinda: $N\ X\ Y\ K$
- $(2+j)$-tā ($0 \leq j \leq N-2$) rinda: $U[j]\ V[j]\ W[j]$

Piemēra vērtētājprogramma katram scenārijam izvada vienu rindu šādā formātā:

- 1. rinda: `max_score` atgrieztā vērtība