International Olympiad in Informatics
2013
6-13 July 2013
Brisbane, Australia

dreaming

Latvian — 1.0

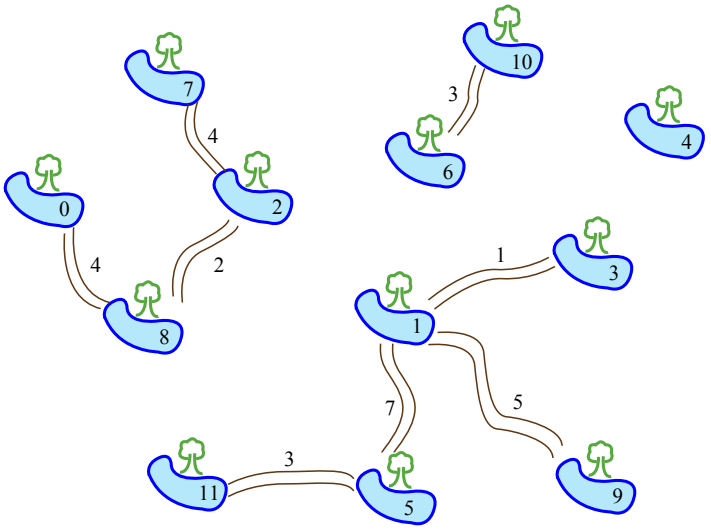
Šis notikums notika ļoti sen, kad pasaule bija jauna un par IOI neviens vēl nesapņoja.

Kādā zemē ir N ūdenstilpes, kas sanumurētas ar skaitļiem $0, ..., N - 1$ un šajā valstī dzīvo Čūska. Ir arī M divvirzienu *kanāli*, kas katrs savieno divas ūdenstilpes, un pa kuriem Čūska var pārvietoties. Katras divas ūdenstilpes savieno (tieši vai netieši) ne vairāk kā viens ceļš (kanālu virkne), bet dažas ūdenstilpes var nebūt savienotas savā starpā (tātad, $M \leq N-1$). Pārvietošanās pa kanāliem Čūskai prasa zināmu dienu skaitu: dažādiem kanāliem dienu skaits var būt atšķirīgs.

Čūskas draugs Ķengurs vēlas uzbūvēt $N - M - 1$ jaunus kanālus, tā lai Čūska varētu nokļūt no jebkuras ūdenstilpes uz jebkuru citu. Ķengurs var uzbūvēt kanālu starp jebkurām divām ūdenstilpēm un jebkura Ķengura uzbūvētā kanāla veikšanai Čūskai būs nepieciešamas L dienas.

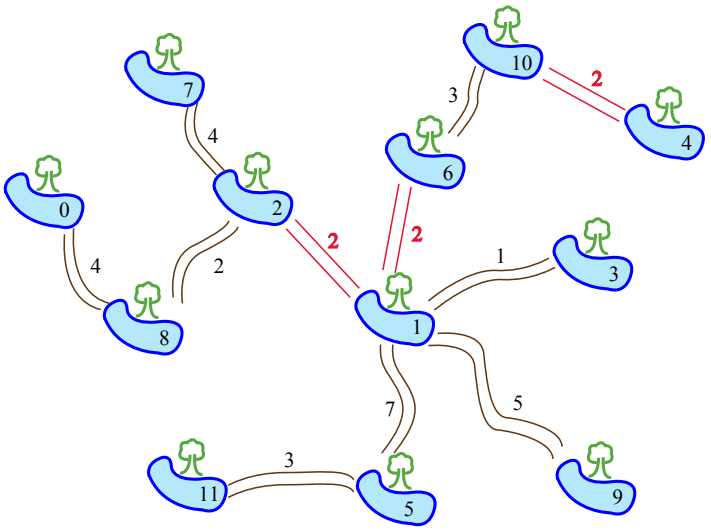
Ķengurs vēlas, lai Čūskai būtu jāveic pēc iespējas īsāki ceļojumi. Ķengurs būvēs jaunos kanālus tā, lai garākais ceļojums starp jebkurām divām ūdenstilpēm aizņemtu pēc iespējas mazāk dienu. Palīdziet Ķenguram un Čūskai noskaidrot garākā ceļojuma laiku starp divām ūdenstilpēm pēc tam, kad Ķengurs būs uzbūvējis jaunos kanālus!

Piemēri



Augstāk redzamajā attēlā ir $N = 12$ ūdenstilpes un $M = 8$ kanāli. Pieņemsim, ka $L = 2$, tātad katra jaunā kanāla veikšanai Čūskai būs nepieciešamas 2 dienas. Ķengurs var uzbūvēt trīs jaunus kanālus:

- starp ūdenstilpēm 1 un 2;
- starp ūdenstilpēm 1 un 6;
- starp ūdenstilpēm 4 un 10.



Augstāk redzamajā attēlā redzams kanālu tīkls pēc jauno kanālu izbūves. Garākais ceļojuma laiks ir 18 dienas, starp ūdenstilpnēm 0 un 11. Lai kā arī Ķengurs būtu būvējis kanālus, šis ir īsākais iespējamais garākā ceļojuma laiks, jo vienmēr būs divas ūdenstilpes starp kurām Čūskai jāceļo 18 dienas vai ilgāk.

Implementācija

Jums jāiesūta fails kurā implementēta funkcija `travelTime()`:

Jūsu procedūra: `travelTime()`

C/C++

```
int travelTime(int N, int M, int L,
        int A[], int B[], int T[]);
```

Pascal

```
function travelTime(N, M, L : LongInt;
        var A, B, T : array of LongInt) : LongInt;
```

Apraksts

Šai funkcijai jāaprēķina ilgākā ceļojuma laiks (dienās) starp jebkurām divām ūdenstilpēm, pieņemot, ka Ķengurs ir uzbūvējis $N - M - 1$ kanālus tā, lai visas ūdenstilpes būtu savienotas un ilgākā ceļojuma laiks ir pēc mazākais iespējamais.

Parametri

- `N` : Ūdenstilpju skaits.
- `M` : Jau esošo kanālu skaits.
- `L` : Dienu skaits, kas nepieciešams Čūskai, lai veiktu jaunu kanālu.
- `A` , `B` un `T` : Masīvi ar garumu `M` , kuros aprakstīti jau esošo kanālu galapunkti un pārvietošanās ilgums pa tiem, `i` -tais kanāls savieno ūdenstilpes ar numuriem `A[i-1]` un `B[i-1]` , un pārvietošanās pa to jebkurā virzienā aizņem `T[i-1]` dienas.
- *Rezultāts*: Garākā ceļojuma laiks starp divām ūdenstilpēm, kā aprakstīts augstāk.

Piemēra sesija

Zemāk esošajā sesijā aprakstīts augstāk redzamais piemērs:

Parameter	Value
N	`12`
M	`8`
L	`2`
A	`[0, 8, 2, 5, 5, 1, 1, 10]`
B	`[8, 2, 7, 11, 1, 3, 9, 6]`
T	`[4, 2, 4, 3, 7, 1, 5, 3]`
Returns	`18`

Ierobežojumi

- Laika ierobežojums: 1 sekunde
- Atmiņas ierobežojums: 64 MiB
- $1 \le N \le 100,000$
- $0 \le M \le N - 1$
- $0 \le A[i], B[i] \le N - 1$
- $1 \le T[i] \le 10,000$
- $1 \le L \le 10,000$

Apakšuzdevumi

Apakšuzdevums	Punkti	Papildus ievaddatu ierobežojumi
1	14	$M = N - 2$, un katrai ūdenstilpei ir tieši viens vai divi tai pievienoti sākotnējie kanāli. Citiem vārdiem, ir divas ūdenstilpju kopas (sakarīgas) un katrā kopā kanāli veido ceļu.
2	10	$M = N - 2$ un $N \leq 100$
3	23	$M = N - 2$
4	18	Katrai ūdenstilpei ir pievienots ne vairāk kā viens sākotnējais kanāls.
5	12	$N \leq 3{,}000$
6	23	*(Nav papildus ierobežojumu)*

Eksperimentēšana

Jūsu datorā esošais piemēru vērtētājs ielasīs ievaddatus no `dreaming.in` faila, šādā formātā:

- 1. rinda: `N M L`
- rindas 2., …, $(M + 1)$.: `A[i] B[i] T[i]`

Augstāk esošais piemērs jāapraksta šādā veidā:

```
12 8 2
0 8 4
8 2 2
2 7 4
5 11 3
5 1 7
1 3 1
1 9 5
10 6 3
```

Piezīmes par valodām

C/C++	Jums jāiekļauj `#include "dreaming.h"`.
Pascal	Jums jādefinē `unit Dreaming`. Visi masīvi tiek numurēti sākot no `0` (nevis `1`).

Iepazīstieties ar risinājumu piemēru šabloniem uz jūsu datora.