



International Olympiad in Informatics 2013

6-13 July 2013
Brisbane, Australia

dreaming

Bosanski — 1.0

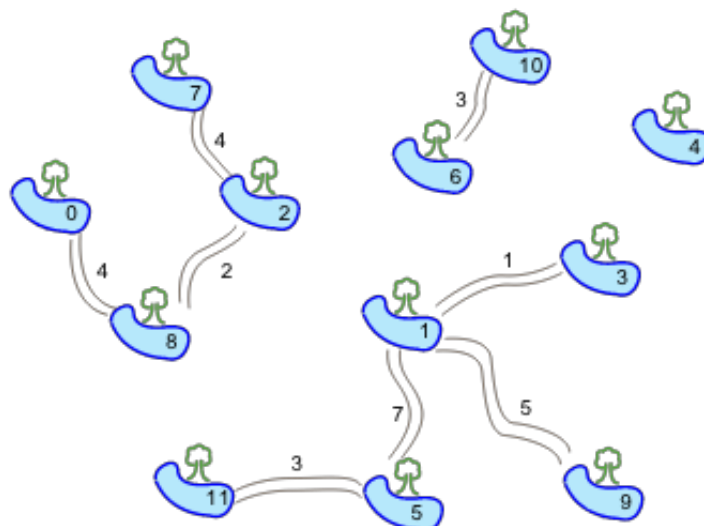
Ova priča smještena je nekada davno, dok je svijet još bio u povoju a IOI nije bio ni zamišljen.

Zmija živi u zemlji u kojoj ima N bazena označenih brojevima $0, \dots, N - 1$. Postoji M dvosmjernih *staza* duž kojih se zmija može kretati između parova bazena. Svaka dva bazena spojena su (direktno ili indirektno) najviše jednim putem (nizom staza), mada određeni parovi bazena uopšte ne moraju biti spojeni (odnosno $M \leq N-1$). Zmiji je potreban određeni broj dana da pređe neku stazu: ovaj broj može biti različit za svaku stazu.

Zmijin prijatelj, Kengur, želi napraviti $N - M - 1$ novih staza tako da se Zmija može kretati između bilo koja dva bazena. Kengur može napraviti stazu između bilo koja dva bazena, a Zmiji će trebati tačno L dana da pređe svaku novu stazu.

Pored toga, Kengur želi Zmijina putovanja skratiti koliko god je to moguće. Kengur će praviti nove staze tako da je najduže vrijeme putovanja između bilo koja dva bazena što je moguće kraće. Pomozite Kenguru i Zmiji da odrede koliko je ovo najduže vrijeme putovanja nakon što Kengur napravi nove staze na opisani način.

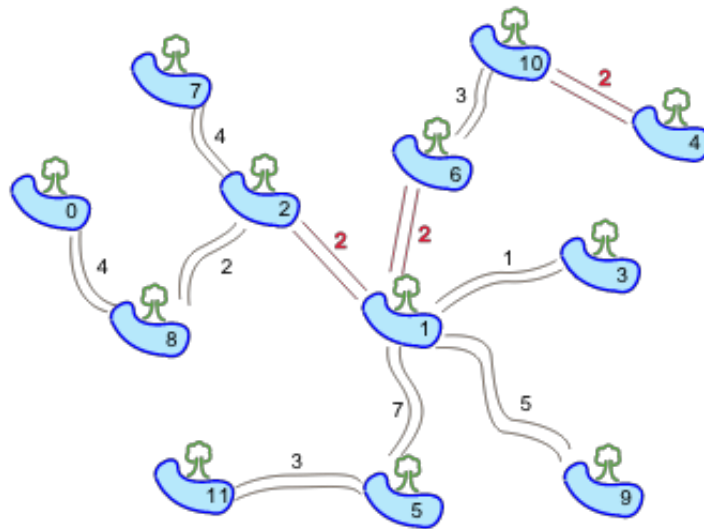
Primjeri



Na slici iznad, nalazi se $N = 12$ bazena i $M = 8$ staza. Neka je $L = 2$, odnosno Zmiji su potrebna dva dana da pređe svaku novu stazu. Tada bi Kengur mogao odlučiti da napravi tri nove staze:

- između bazena 1 i 2;

- između bazena 1 i 6;
- između bazena 4 i 10.



Na slici iznad prikazan je krajnji skup staza. Najduže vrijeme putovanja je 18 dana (između bazena 0 i 11). Ovo je najbolji mogući rezultat—bez obzira na koji način Kengur izgradi nove staze, postojat će neka dva bazena između kojih će Zmija putovati barem 18 dana.

Implementacija

Trebate submitovati file koji implementira funkciju `travelTime()` kako slijedi:

Vaša funkcija: `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;`

Opis

Ova funkcija treba da izračuna najduže vrijeme putovanja (izraženo u danima) između bilo koja dva bazena, uz pretpostavku da je Kengur izgradio `N - M - 1` novih staza na način da su svi bazeni povezani i da je ovo najduže vrijeme putovanja što je moguće manje.

Parametri

- `N`: Broj bazena.
- `M`: Broj staza koje već postoje.
- `L`: Vrijeme u danima potrebno Zmiji da pređe novu stazu.

- A , B i T : Nizovi dužine M koji određuju krajeve već postojećih staza i vrijeme putovanja po svakoj od njih, na način da i -ta staza spaja bazene $A[i-1]$ i $B[i-1]$, te da je Zmiji potrebno $T[i-1]$ dana da pređe stazu u bilo kojem smjeru.
- *Vraća*: Najduže vrijeme putovanja između bilo koja dva bazena, kako je opisano ranije.

Primjer izvršavanja

Sljedeći primjer izvršavanja odnosi se na ranije navedeni primjer:

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

Ograničenja

- Vremensko ograničenje: 1 second
- Memorijsko ograničenje: 64 MiB
- $1 \leq N \leq 100,000$
- $0 \leq M \leq N - 1$
- $0 \leq A[i], B[i] \leq N - 1$
- $1 \leq T[i] \leq 10,000$
- $1 \leq L \leq 10,000$

Podzadaci

Podzadatak	Poena	Dodatna ograničenja na ulazne podatke
1	14	$M = N - 2$, i postoji tačno jedna ili tačno dvije već postojeće staze koje vode iz svakog od bazena. Drugim riječima, postoje dva skupa povezanih bazena, i u svakom od njih staze formiraju put koji se nikada ne grana.
2	10	$M = N - 2$ i $N \leq 100$
3	23	$M = N - 2$
4	18	Postoji najviše jedna već postojeća staza koja vodi iz svakog od bazena.
5	12	$N \leq 3,000$
6	23	(Bez dodatnih ograničenja)

Testiranje

Grader koji je dostupan na Vašem računaru će ulazne podatke čitati iz file-a `dreaming.in`, u sljedećem formatu:

- linija 1: `N M L`
- linije 2, ..., `M + 1`: `A[i] B[i] T[i]`

Ilustracije radi, ranije opisani primjer je zapisan u sljedećem formatu:

```
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
```

Napomene za programske jezike

C/C++ Treba uraditi `#include "dreaming.h"`.

Pascal Trebate definisati `unit Dreaming`. Svi nizovi su numerisani sa početkom u `0` (ne `1`).

Pogledati primjer rješenja na Vašem računaru za predložene primjere.