



International Olympiad in Informatics

2013

6-13 July 2013

Brisbane, Australia

dreaming

Turkish — 1.0

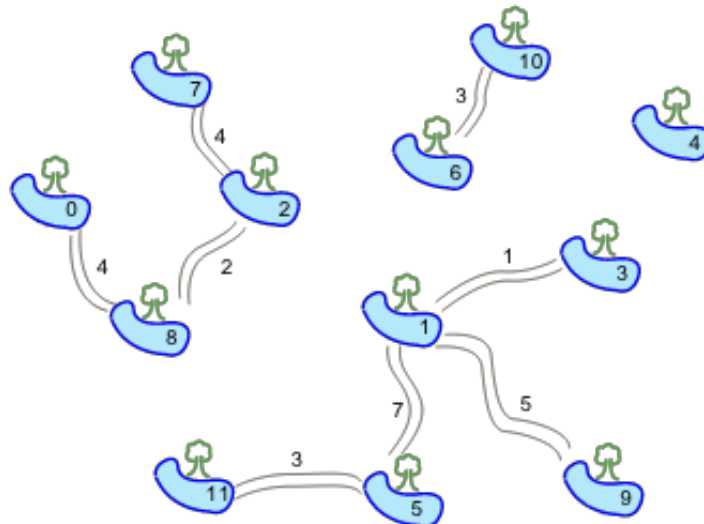
Bu masal uzun yıllar önce dünya yeni kurulmuş, IOI ise henüz hayal bile değilken yaşanmıştır.

İçinde N tane (numaralandırma $0, \dots, N-1$) su kuyusu bulunan ülkede bir Yılan yaşamaktadır. Su kuyusu çiftlerini birbirine bağlayan M adet çiftyönlü *yol* bulunmaktadır. Yılan, bu yollar üzerinde seyahat edebilmektedir. Herhangi iki su kuyusu çifti en fazla bir adet yol sıralısı ile birbirine bağlıdır, bazı su kuyuları arasında ise yol olmayabilir (yani $M \leq N-1$). Her bir yol için, Yılan'ın o yolu katetmesi (seyahati) belli bir gün almaktadır ve bu süre her bir yol için farklı farklı olabilir.

Yılan'ın arkadaşı Kanguru $N - M - 1$ tane yeni yol inşa etme niyetindedir, öyle ki Yılan herhangi iki su kuyusu arasında seyahat edebilsin. Kanguru seçtiği herhangi iki su kuyusu arasında yeni yol (kenar) inşa edebilir, bu yeni yolların her birini Yılan L günde seyahat eder.

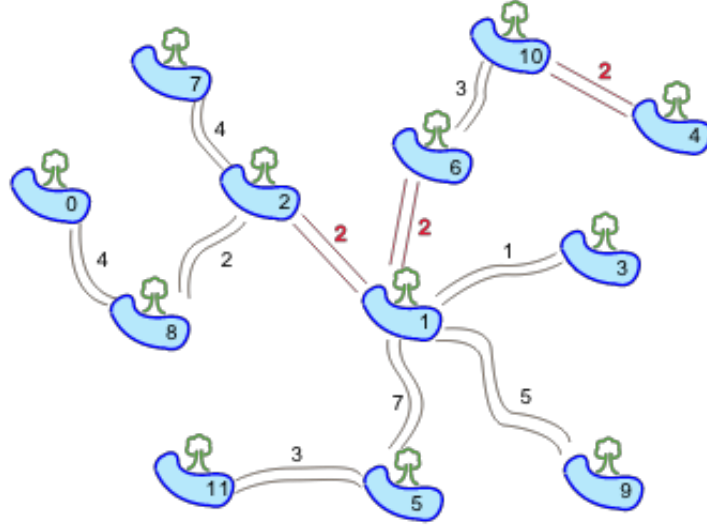
İlave olarak Kanguru, Yılan'ın seyahatlerinin olabildiğince hızlı olmasını istemektedir. Kanguru yeni inşa edeceği yolları öyle seçer ki herhangi iki su kuyusu arasındaki en uzun seyahat süresi en kısa olsun. Kanguru'nun bu şekilde yol inşa ettiğini siz biliyorsunuz, bundan hareketle Kanguru'ya herhangi iki su kuyusu arasındaki en uzun seyahat süresini bulmasında yardımcı olun.

Örnekler



Yukarıdaki resimde $N = 12$ su kuyusu ve $M = 8$ yol (kenar) vardır. $L = 2$ olduğunda, Yılan yeni inşa edilen her bir yolu 2 günde seyahat edebilir. Böylece Kanguru aşağıdaki üç yeni yolu inşa edebilir:

- su kuyuları 1 ve 2 arasında;
- su kuyuları 1 ve 6 arasında;
- su kuyuları 4 ve 10 arasında.



Yukarıdaki resim en son yol durumunu göstermektedir. Burada en uzun seyahat süresi (su kuyuları 0 ve 11 arasında) 18 gündür. Bu olabilecek en küçük sonuçtur, yani Kanguru'nun başka çözümleri en az 18 veya daha fazla seyahat süresi gerektirecek su kuyusu çiftleri olmasını gerektirecekti.

Gerçekleştirim

Aşağıdaki şekilde tanımlanmış `travelTime()` adında fonksiyon içeren bir dosya göndermeniz gerekmektedir:

Fonksiyonunuz: `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;
```

Tanım

Bu fonksiyon, Kanguru'nun tüm su kuyusu çiftleri arasında en uzun seyahat süresini en küçük yapacak şekilde $N - M - 1$ (tüm su kuyuları birbirine bir yol sıralısı ile bağlı) adet yeni yol inşa ettiği kabulü altında, herhangi iki su kuyusu çifti arasındaki (gün bazında) en uzun seyahat süresini hesaplamalıdır.

Parametreler

- N : Su kuyusu sayısı.
- M : Başlangıçta mevcut olan yol (kenar) sayısı.
- L : Yılan'ın iki su kuyusu arasında yeni inşa edilen bir yolu seyahat süresi (gün olarak).
- A , B ve T : Önceden mevcut olan i .nci yol $A[i-1]$ ve $B[i-1]$ su kuyuları arasındadır, ve bu yolun (kenar) seyahat süresi $T[i-1]$ gündür.
- *Dönen Değer*: Yukarıda anlatıldığı gibi hesaplanacak herhangi iki su kuyusu arasındaki en uzun seyahat süresi.

Örnekteki Durum

Aşağıdaki durum yukarıda verilen örneği açıklar:

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

Kısıtlar

- Süre sınırı: 1 saniye
- Hafıza sınırı: 64 MiB
- $1 \leq N \leq 100,000$
- $0 \leq M \leq N - 1$
- $0 \leq A[i], B[i] \leq N - 1$
- $1 \leq T[i] \leq 10,000$
- $1 \leq L \leq 10,000$

Altgörevler

Altgörev	Puan	İlave Girdi Kısıtları
1	14	$M = N - 2$, ve ilk durumda her bir su kuyusuna bağlı tam olarak bir ya da iki yol vardır. Yani, kendi içinde birbirlerine bağlı iki ayrı su kuyusu kümesi vardır, ve bu kümelerin herbirinde hiç bir yolda dallanma yoktur.
2	10	$M = N - 2$ ve $N \leq 100$
3	23	$M = N - 2$
4	18	İlk durumda her bir su kuyusuna bağlı en fazla bir adet yol (kenar) vardır.
5	12	$N \leq 3,000$
6	23	(Kısıt yok)

Test etme

Bilgisayarınızdaki örnek grader `dreaming.in` adlı girdi dosyasını okumaktadır, bu dosya şu formatta olmalıdır:

- satır 1: `N M L`
- satırlar 2, ..., $M + 1$: `A[i] B[i] T[i]`

Örneğin, yukarıdaki örnek şu formatta verilmelidir:

```
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
```

Programlama Dili Notları

C/C++ `#include "dreaming.h"` dosyasını include etmelisiniz.

Pascal Önce `unit Dreaming` i tanımlamalısınız. Bütün dizi indisleri 0 dan başlar.

Örnekler için bilgisayarınızdaki çözüm şablonlarını inceleyebilirsiniz.

