



## International Olympiad in Informatics 2013

6-13 July 2013

Brisbane, Australia

**dreaming**

Romana — 1.1

Această poveste se petrece cu mult timp în urmă, atunci când lumea era nouă și IOI nu fusese încă visat.

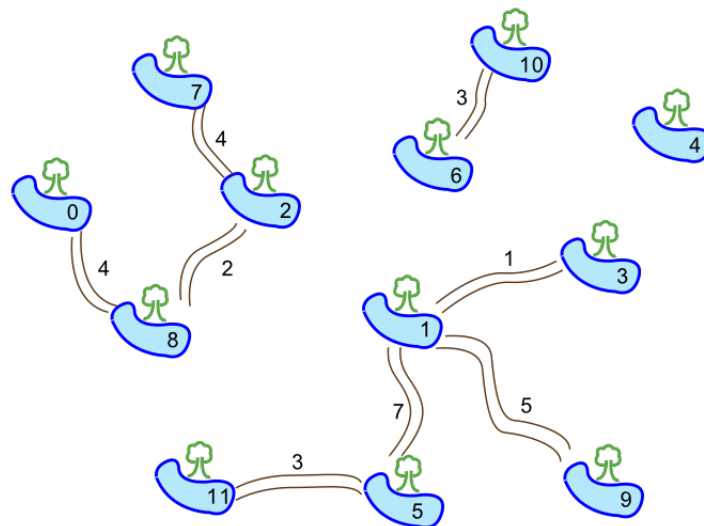
Serpent trăiește într-o lume în care există  $N$  băltoace, numerotate  $0, \dots, N-1$ . Există  $M$  poteci bidirecționale, care unesc perechi de băltoace, pe care Serpent poate călători. Fiecare pereche de băltoace este conectată (direct sau indirect) de cel mult o secvență de poteci, deși unele perechi de băltoace pot să nu fie conectate deloc (deci,  $M \leq N-1$ ). Fiecare potecă necesită un număr de zile pentru ca Serpent să o parcurgă: acest număr poate fi diferit pentru fiecare potecă în parte.

Prietenul lui Serpent, Kangaroo, dorește să construiască  $N - M - 1$  noi poteci, astfel încât să fie posibil ca Serpent să călătorească între oricare pereche de băltoace. Kangaroo poate crea poteci între oricare pereche de băltoace, iar fiecare potecă creată de Kangaroo va dura  $L$  zile pentru ca Serpent să o călătorească.

În plus, Kangaroo dorește să facă călătoriile lui Serpent pe cât de scurte posibil. Kangaroo va construi noi poteci astfel încât timpul de parcurgere al celui mai lung drum între oricare pereche de băltoace să fie cât mai mic posibil. Ajutați-l pe Kangaroo și Serpent să determine cel mai lung timp de călătorie între oricare pereche de băltoace, după ce Kangaroo a construit potecile noi în stiuł descris.

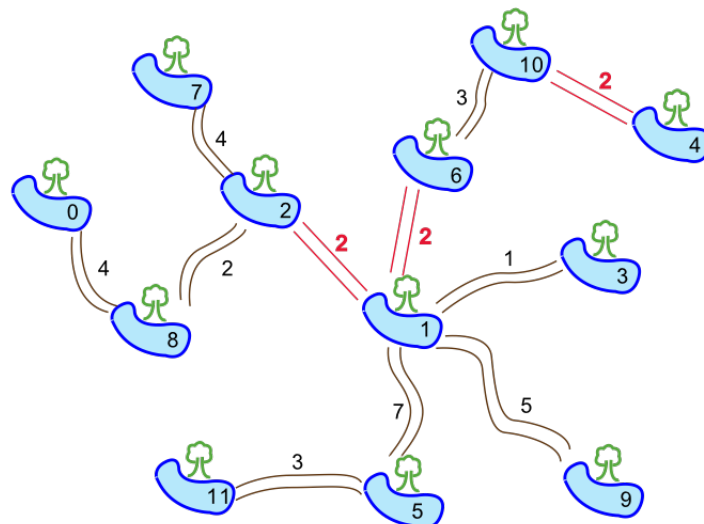
---

## Exemple



În imaginea de mai sus, sunt  $N = 12$  băltoace și  $M = 8$  poteci. Presupunând că  $L = 2$  atunci fiecare potecă nouă va fi călătorită de Serpent în 2 zile. Atunci Kangaroo poate construi trei noi poteci:

- între băltoacele 1 și 2;
- între băltoacele 1 și 6;
- între băltoacele 4 și 10.



Imaginea de mai sus arată setul final de poteci. Cel mai lung timp de călătorie este de 18 zile, între băltoacele 0 și 11. Acesta este cel mai mic rezultat posibil - oricum Kangaroo ar construi potecile, va exista o pereche de băltoace care să necesite ca Serpent să călătorească 18 zile sau mai mult.

---

## Implementare

Va trebui să submitați un fișier ce implementează funcția `travelTime()`, după cum urmează:

### Funcția voastră: `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;
```

### Descriere

Această funcție trebuie să calculeze cel mai mare timp de călătorie (măsurat în zile) între orice pereche de băltoace, presupunând că Kangaroo a adăugat  $N - M - 1$  poteci astfel încât toate băltoacele sunt conectate și timpul maxim de călătorie este cel mai mic posibil.

### Parametrii

- $N$ : Numărul de băltoace.
- $M$ : Numărul de poteci ce există deja.
- $L$ : Timpul în zile care care îi trebuie lui Serpent să călătorească pe o potecă nouă.
- $A$ ,  $B$  and  $T$ : Vectori de lungime  $M$  ce specifică capetele și timpul de parcurgere pentru fiecare potecă existentă, astfel încât a  $i$  a potecă unește băltoacele  $A[i-1]$  și  $B[i-1]$ , și necesită  $T[i-1]$  zile pentru a fi călătorită în orice direcție.
- *Returnează*: Cel mai mare timp de călătorie între orice pereche de băltoace, așa cum este descris mai sus.

---

## Exemplu de scenariu

Următorul scenariu descrie exemplul de mai sus:

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

## Constrângeri

- Limită de timp: 1 secundă
- Limită de memorie: 64 MiB
- $1 \leq N \leq 100,000$
- $0 \leq M \leq N - 1$
- $0 \leq A[i], B[i] \leq N - 1$
- $1 \leq T[i] \leq 10,000$
- $1 \leq L \leq 10,000$

## Subtask-uri

Subtask	Puncte	Constrângeri suplimentare
1	14	$M = N - 2$ , și există exact una sau două poteci pre-existente ce pleacă din fiecare băltoacă. Cu alte cuvinte, există două seturi de băltoace conectate, și în fiecare set potecile formează un drum ce nu se desprinde.
2	10	$M = N - 2$ și $N \leq 100$
3	23	$M = N - 2$
4	18	Există cel mult o singură potecă pre-existentă ce pleacă din fiecare băltoacă.
5	12	$N \leq 3,000$
6	23	( <i>Nimic</i> )

---

## Testare

Grader-ul de pe computerul vostru va citi input-ul din fișierul `dreaming.in`, care trebuie să fie în următorul format:

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

În acest fel, exemplul de mai sus trebuie să fie dat în următorul format:

```
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
```

---

## Note de limbaj

C/C++ Trebuie să faceți `#include "dreaming.h"`.

Pascal Trebuie să definiți `unit Dreaming`. Toți vectorii sunt indexați de la 0 (nu de la 1).