



International Olympiad in Informatics 2013

6-13 July 2013

Brisbane, Australia

dreaming

French — 1.0

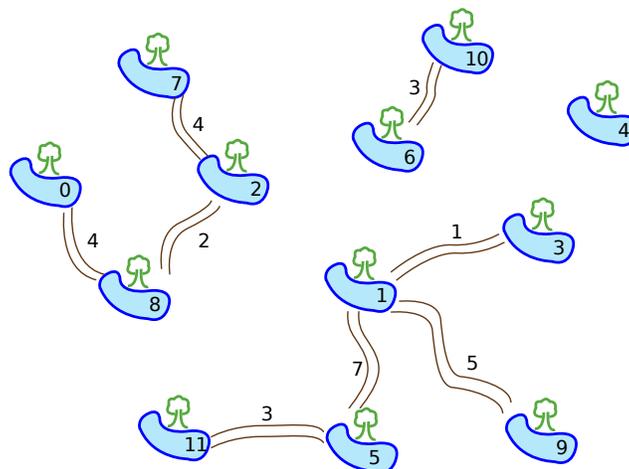
Cette histoire se passe il y a très longtemps, lorsque le monde était nouveau et que les IOI n'avaient pas encore été imaginées.

Serpent vit sur un terrain percé de N billabongs (trous d'eau), numérotés de 0 à $N-1$. Il y a M pistes à double sens, reliant des paires de billabongs, que Serpent peut traverser. Chaque paire de billabongs est connectée (directement ou indirectement) par au plus une séquence de pistes, mais certaines paires de billabongs peuvent ne pas être connectées du tout (donc $M \leq N-1$). Serpent prend un certain nombre de jours pour traverser chaque piste : ce nombre peut être différent pour chacune d'elles.

Kangourou, l'ami de Serpent, veut créer $N-M-1$ nouvelles pistes, pour que Serpent puisse voyager entre toute paire de billabongs. Kangourou peut créer de nouvelles pistes entre n'importe quelles paires de billabongs. Serpent traversera ces nouvelles pistes en L jours chacune.

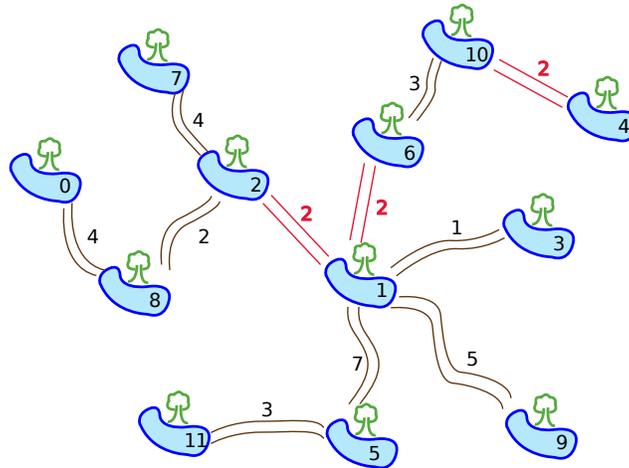
De plus, Kangourou veut que Serpent voyage le plus rapidement possible. Kangourou va créer les nouvelles pistes de sorte que la durée pour voyager entre toute paire de billabongs soit la plus courte possible. Aidez Kangourou et Serpent à déterminer le plus long temps de trajet entre toute paire de billabongs, une fois que Kangourou aura construit les nouvelles pistes de cette manière.

Exemples



Sur l'image ci-dessus, il y a $N = 12$ billabongs et $M = 8$ pistes. Supposons que $L = 2$, c'est-à-dire que Serpent traversera toute nouvelle piste en 2 jours. Kangourou peut créer trois nouvelles pistes :

- entre les billabongs 1 et 2,
- entre les billabongs 1 et 6,
- entre les billabongs 4 et 10.



L'image ci-dessus montre l'ensemble final de pistes. Le trajet le plus long prend 18 jours, entre les billabongs 0 et 11. C'est le plus petit résultat possible — quelle que soit la façon dont Kangourou construit les pistes, il y aura des paires de billabongs qui demanderont à Serpent 18 jours de voyage ou plus.

Implémentation

Vous devez soumettre un fichier qui implémente la fonction `travelTime()`, comme suit :

Votre fonction : `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;
```

Description

Cette fonction doit calculer la durée du voyage le plus long (mesurée en jours) entre toute paire de billabongs, en supposant que Kangourou a ajouté $N - M - 1$ pistes de sorte que tous les billabongs soient connectés et que cette durée du voyage le plus long soit la plus courte possible.

Paramètres

- `N` : le nombre de billabongs.
- `M` : le nombre de pistes existantes au départ.
- `L` : la durée en jours que Serpent prend pour traverser une nouvelle piste.
- `A`, `B` et `T` : tableaux de taille `M` qui spécifient les extrémités et la durée de parcours de chaque piste existante, de sorte que la i -ième piste relie les billabongs `A[i-1]` et `B[i-1]` et prend `T[i-1]` jours à traverser quel que soit le sens.
- *Retourne* : la plus longue durée de voyage parmi toutes les paires de billabongs, comme décrit ci-dessus.

Exemple de session

La session suivante décrit l'exemple ci-dessus :

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

Contraintes

- Limite de temps : 1 seconde
- Limite de mémoire : 64 Mio
- $1 \leq N \leq 100\,000$
- $0 \leq M \leq N - 1$
- $0 \leq A[i], B[i] \leq N - 1$
- $1 \leq T[i] \leq 10\,000$
- $1 \leq L \leq 10\,000$

Sous-tâches

Sous-tâche	Points	Contraintes d'entrée supplémentaires
1	14	$M = N - 2$ et il y a exactement une ou deux pistes existantes partant de chaque billabong. En d'autres mots, il y a deux ensembles de billabongs connectés et dans chaque ensemble les pistes forment une chaîne.
2	10	$M = N - 2$ et $N \leq 100$
3	23	$M = N - 2$
4	18	Il y a au plus une piste existante partant de chaque billabong.
5	12	$N \leq 3\,000$
6	23	(Aucune)

Expérimentation

L'évaluateur fourni sur votre ordinateur lira l'entrée dans le fichier `dreaming.in`, qui doit être au format suivant :

- ligne 1 : `N M L`
- lignes 2 à `M + 1` : `A[i] B[i] T[i]`

Par exemple, l'exemple ci-dessus doit être fourni au format suivant :

```
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
```

Remarques pour chaque langage

C/C++ Vous devez utiliser `#include "dreaming.h"`.

Pascal Vous devez utiliser l'unité `unit Dreaming`. Tous les tableaux sont numérotés à partir de `0` (et non `1`).

Prenez exemple sur les modèles de solution fournis sur votre machine.