



## International Olympiad in Informatics 2013

6-13 July 2013

Brisbane, Australia

# dreaming

Greek — 1.0

Αυτή η ιστορία συνέβη πολλά χρόνια πριν, στη γένεση του κόσμου και δεν είχαμε καν ονειρευτεί την IOI

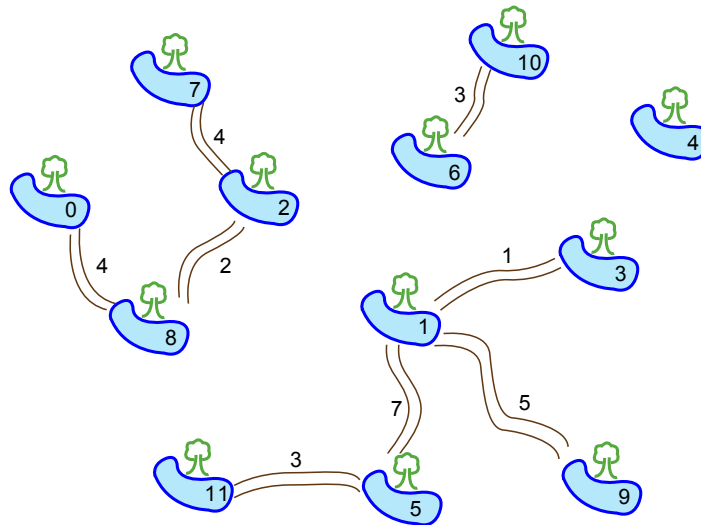
Ο Serpent κατοικεί σε μια περιοχή με  $N$  λακκούβες νερού αριθμημένες  $0, \dots, N - 1$ . Υπάρχουν  $M$  μονοπάτια, διπλής κατεύθυνσης, που συνδέουν ζεύγη από λακκούβες, τα οποία μπορεί να διασχίσει ο Serpent. Κάθε ζεύγος από λακκούβες συνδέεται (άμεσα ή έμμεσα) με το πολύ μια ακολουθία μονοπατιών, αν και μερικά ζεύγη από λακκούβες μπορεί να μην συνδέονται καθόλου (επομένως,  $M \leq N - 1$ ). Κάθε μονοπάτι ο Serpent το διασχίζει σε ένα πλήθος ημερών: αυτός ο αριθμός μπορεί να είναι διαφορετικός για κάθε μονοπάτι.

Ο φίλος του Serpent, ο Kangaroo, θέλει να φτιάξει  $N - M - 1$  νέα μονοπάτια, έτσι που να μπορεί ο Serpent να ταξιδέψει μεταξύ οποιουδήποτε ζεύγους από λακκούβες. Ο Kangaroo μπορεί να δημιουργήσει μονοπάτια μεταξύ οπουδήποτε ζεύγους από λακκούβες, και κάθε μονοπάτι που δημιουργεί ο Kangaroo θα απαιτηθούν  $L$  ημέρες για τον Serpent για να το διανύσει.

Επιπλέον, ο Kangaroo θέλει τα ταξίδια του Serpent να είναι όσο το δυνατόν πιο γρήγορα. Ο Kangaroo θα φτιάξει τα νέα μονοπάτια έτσι που ο μεγαλύτερος χρόνος της διαδρομής μεταξύ δυο οποιονδήποτε λακκουβών να είναι όσο το δυνατόν μικρή. Βοηθήστε τον Kangaroo και τον Serpent να προσδιορίσουν την μεγαλύτερη σε χρόνο διαδρομή μεταξύ δυο λακκουβών, αφού ο Kangaroo έχει κατασκευάσει τις νέες διαδρομές με αυτόν τον τρόπο.

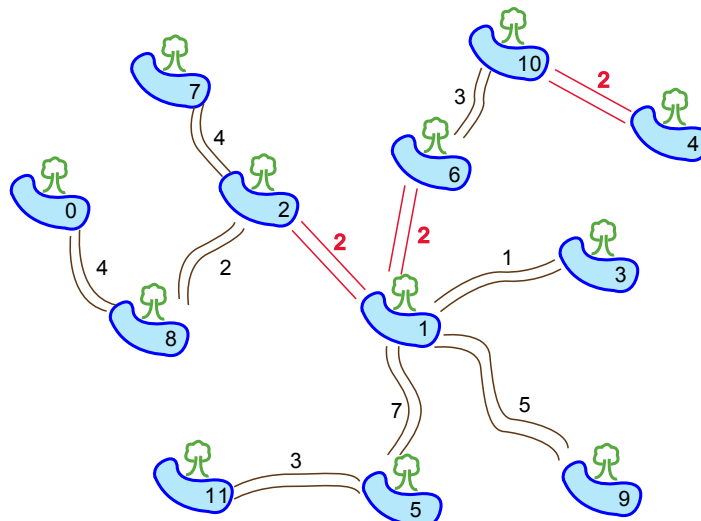
---

## Παραδείγματα



Στην παραπάνω εικόνα υπάρχουν  $N = 12$  λακκούβες και  $M = 8$  μονοπάτια. Υποθέστε ότι  $L = 2$ , έτσι που για κάθε νέα διαδρομή θα χρειασθεί στον Serpent 2 ημέρες ταξιδιού. Σε αυτήν την περίπτωση θα μπορούσε ο Kangaroo να κατασκευάσει τρεις νέες διαδρομές:

- μεταξύ των λακκουβών 1 και 2;
- μεταξύ των λακκουβών 1 και 6;
- μεταξύ των λακκουβών 4 και 10.



Η παραπάνω εικόνα δείχνει το τελικό σύνολο των διαδρομών. Η μεγαλύτερη σε χρόνο διαδρομή είναι 18 ημέρες, μεταξύ των λακκουβών 0 and 11. Αυτό είναι το μικρότερο δυνατό αποτέλεσμα—άσχετα με τον τρόπο που κτίζει τις διαδρομές ο Kangaroo, θα υπάρξει ένα ζεύγος από λακκούβες που για να πάει από την μια στην άλλη ο Serpent χρειάζεται 18 ή περισσότερες ημέρες.

## Υλοποίηση

Θα πρέπει να υποβάλετε ένα αρχείο που να υλοποιεί την συνάρτηση `travelTime()`, όπως παρακάτω:

## Η συνάρτησή σας: `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;
```

## Περιγραφή

Η συνάρτηση αυτή θα πρέπει να υπολογίζει τον μέγιστο χρόνο ταξιδιού (μετρημένο σε ημέρες) μεταξύ οποιουδήποτε ζεύγους λακκουβών, υποθέτοντας ότι ο Kangaroo έχει προσθέσει  $N - M - 1$  μονοπάτια με τέτοιο τρόπο ώστε όλες οι λακκούβες συνδέονται και αυτός ο μέγιστος χρόνος ταξιδιού είναι όσο το δυνατόν μικρός.

## Παράμετροι

- `N`: Το πλήθος των λακκουβών.
- `M`: Το πλήθος των μονοπατιών που ήδη υπάρχουν.
- `L`: Ο χρόνος σε ημέρες που χρειάζεται ο Serpent για να διασχίσει κατά μήκος ενός μονοπατιού.
- `A`, `B` και `T`: Πίνακες μεγέθους `M` που προσδιορίζουν τα άκρα και τον χρόνο διάσχισης κάθε προϋπάρχοντος μονοπατιού έτσι ώστε, έτσι ώστε το  $i$  th μονοπάτι συνδέει τις λακκούβες `A[i-1]` και `B[i-1]`, και απαιτεί `T[i-1]` ημέρες για να το διασχίσει σε οποιαδήποτε κατεύθυνση.
- *Επιστρέφει*: Τον μέγιστο χρόνο ταξιδιού μεταξύ οποιουδήποτε ζεύγους λακκουβών, όπως περιγράφεται παραπάνω.

---

## Δείγμα

Το παρακάτω περιγράφει το παραπάνω παράδειγμα:

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

## Περιορισμοί

- Όριο Χρόνου: 1 second
- Όριο μνήμης: 64 MiB (64MB)
- $1 \leq N \leq 100,000$
- $0 \leq M \leq N - 1$
- $0 \leq A[i], B[i] \leq N - 1$
- $1 \leq T[i] \leq 10,000$
- $1 \leq L \leq 10,000$

## Υποπροβλήματα

Υποπρόβλημα	Βαθμοί	Επιπλέον Περιορισμοί Εισόδου
1	14	$M = N - 2$ , και υπάρχουν ακριβώς ένα ή δύο προϋπάρχοντα μονοπάτια από κάθε λακκούβα. Με άλλα λόγια, υπάρχουν δυο σύνολα συνδεδεμένων λακκουβών, και σε κάθε σύνολο τα μονοπάτια δεν σχηματίζουν κλειστή διαδρομή.
2	10	$M = N - 2$ and $N \leq 100$
3	23	$M = N - 2$
4	18	Υπάρχει το πολύ ένα προϋπάρχον μονοπάτι που φεύγει από κάθε λακκούβα.
5	12	$N \leq 3,000$
6	23	(None)

## Πειραματισμός

Το δείγμα βαθμολογητή στον υπολογιστή σας θα διαβάζει είσοδο από το αρχείο `dreaming.in`, που θα έχει την παρακάτω μορφοποίηση:

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

Για παράδειγμα, για το παραπάνω παράδειγμα τα δεδομένα τα δεδομένα θα πρέπει να δίδονται στην παρακάτω μορφή:

```
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
```

---

## Σημειώσεις για την γλώσσα

**C/C++** Πρέπει να ορίσετε `#include "dreaming.h"`.

**Pascal** Πρέπει να συμπεριλάβετε την `unit Dreaming`. Σε όλους τους πίνακες η αρίθμηση των στοιχείων ξεκινά από `0` (και όχι `1`).

Δείτε τις προτεινόμενες λύσεις στον υπολογιστή σας για παραδείγματα.