



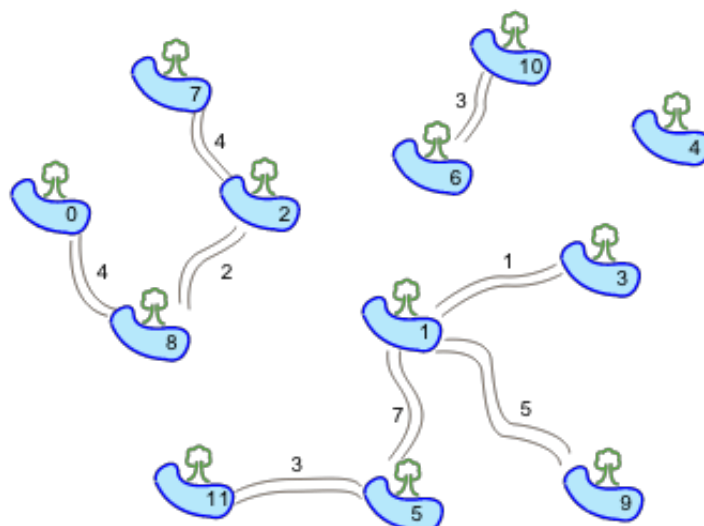
Cerita ini terjadi dahulu kala, ketika dunia masih baru dan IOI belum terbayangkan.

Serpent tinggal di daratan yang memiliki N billabong (lubang air) yg bernomor $0, \dots, N-1$. Terdapat M jalur dua arah (bidirectional trails), menggabungkan pasangan billabong, yang mana Serpent dapat melakukan perjalanan. Setiap pasang billabong terhubung (langsung maupun tidak langsung) oleh paling banyak satu barisan jalur, meskipun beberapa pasang jalur billabong mungkin tidak terhubung sama sekali (sehingga, $M \leq N-1$). Setiap jalur ditempuh dalam sejumlah hari untuk Serpent melakukan perjalanan sepanjang jalur tersebut: jumlah ini mungkin berbeda untuk setiap jalur.

Teman Serpent, Kangaroo, ingin membuat $N-M-1$ jalur baru, sehingga memungkinkan bagi Serpent untuk menempuh perjalanan antara setiap pasangan billabong manapun. Kangaroo dapat membuat jalur baru antara pasangan billabong manapun, dan setiap jalur yang Kangaroo ciptakan akan memakan waktu L hari bagi Serpent untuk melakukan perjalanan sepanjang itu.

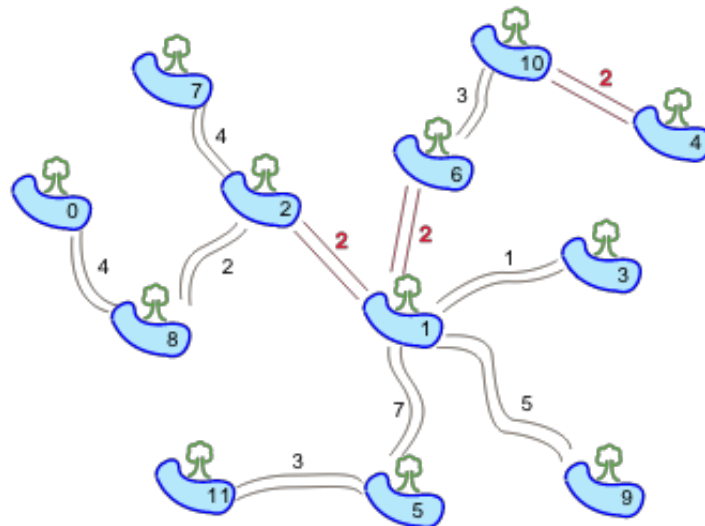
Sebagai tambahan, Kangaroo ingin membuat perjalanan yang secepat mungkin bagi Serpent. Kangaroo akan membangun jalur baru sehingga waktu perjalanan terpanjang antara setiap dua billabong adalah yang sependek mungkin. Bantulah Kangaroo dan Serpent menentukan waktu perjalanan terpanjang antar tiap dua billabong, setelah Kangaroo membangun jalur-jalur baru tersebut dengan cara seperti ini.

Contoh



Pada gambar di atas terdapat $N = 12$ billabong dan $M = 8$ jalur. Andaikan $L = 2$, sehingga setiap jalur baru akan membutuhkan 2 hari untuk Serpent melakukan perjalanan. Maka Kangaroo akan dapat membangun 3 jalur baru:

- antara billabong 1 dan 2;
- antara billabong 1 dan 6;
- antara billabong 4 dan 10.



Gambar di atas menunjukkan himpunan jalur akhir. Waktu perjalanan terpanjang adalah 18 hari, antara billabong 0 dan 11. Ini adalah hasil perpendek yang mungkin--tidak peduli bagaimana Kangaroo membangun jalur, akan terdapat beberapa pasang billabong dimana Serpent akan memerlukan waktu perjalanan selama 18 hari atau lebih.

Implementasi

Anda harus me-submit sebuah file yang mengimplementasikan `travelTime()`, sebagai berikut:

Fungsi Anda: `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;
```

Deskripsi

Fungsi ini akan menghitung waktu perjalanan terbesar (dalam ukuran hari) antara setiap pasang billabong, diasumsikan Kangaroo telah menambahkan $N - M - 1$ jalur sedemikian rupa sehingga seluruh billabong terhubung dan waktu perjalanan terbesar adalah sekecil mungkin.

Parameter

- N : jumlah billabong.
- M : jumlah jalur yang telah ada.
- L : waktu dalam hari yang diperlukan Serpent untuk melakukan perjalan sepanjang jalur baru.
- A , B dan T : array dengan panjang M yang menentukan ujung-ujung jalur dan waktu perjalanan dari setiap jalur yang sudah ada, sedemikian rupa sehingga jalur ke- i menghubungkan billabong $A[i-1]$ dan $B[i-1]$, dan membutuhkan $T[i-1]$ hari untuk melakukan perjalanan di kedua arah.
- *Returns*: waktu perjalanan terbesar antara setiap pasang billabong, seperti dideskripsikan di atas.

Contoh sesi

Sesi berikut ini menunjukkan contoh yang telah diberikan di atas:

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

Batasan

- Batas waktu: 1 detik
- Batas memori: 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

Subtask	Nilai	Batasan masukan tambahan
1	14	$M = N - 2$, dan ada tepat satu atau dua jalur yang sudah ada yang mengarah dari setiap billabong. Dengan kata lain, ada dua himpunan billabong terhubung, dan dalam setiap himpunan tidak terbentuk jalur bercabang.
2	10	$M = N - 2$ dan $N \leq 100$
3	23	$M = N - 2$
4	18	ada paling banyak satu jalur yang sudah ada yang mengarah dari setiap billabong.
5	12	$N \leq 3,000$
6	23	(Tidak Ada)

Eksperimentasi

Contoh grader di komputer anda akan membaca input dari file `dreaming.in`, yang harus mengikuti format:

- baris 1: $N \ M \ L$
- baris 2, ..., $M + 1$: $A[i] \ B[i] \ T[i]$

Misalnya, untuk contoh di atas, input diberikan dalam format berikut:

```
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
```

Catatan Bahasa yang Digunakan

C/C++ Anda harus meng- `#include "dreaming.h"` .

Pascal Anda harus mendefinisikan `unit Dreaming` . Semua array dinomori mulai dari `0` (bukan `1`).

Lihat template solusi di mesin anda sebagai contoh.