



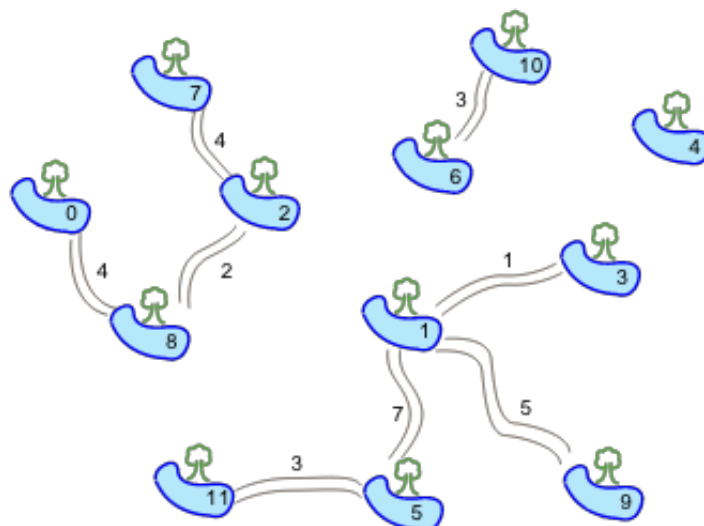
Questa storia si svolge tanto tempo fa, quando il mondo era ancora giovane e le IOI non erano ancora state sognate.

Il signor Serpente vive in una terra che ha  $N$  billabong (pozze d'acqua), numerate  $0, \dots, N - 1$ . Ci sono  $M$  *sentieri* bidirezionali lungo cui Serpente si sposta, ciascuno dei quali collega due billabong. Ogni coppia di billabong è connessa da al più una sequenza di sentieri, ma alcune coppie di billabong potrebbero non essere connesse (quindi  $M \leq N - 1$ ). Serpente impiega un certo numero di giorni per percorrere ciascun sentiero: il numero può essere diverso da sentiero a sentiero.

Canguro, l'amico di Serpente, vuole creare  $N - M - 1$  nuovi sentieri, in modo che per ciascuna coppia di billabong Serpente possa spostarsi da un billabong all'altro. Canguro può creare sentieri tra qualunque coppia di billabong, e Serpente impiega sempre  $L$  giorni a percorrere un qualunque sentiero creato da Canguro.

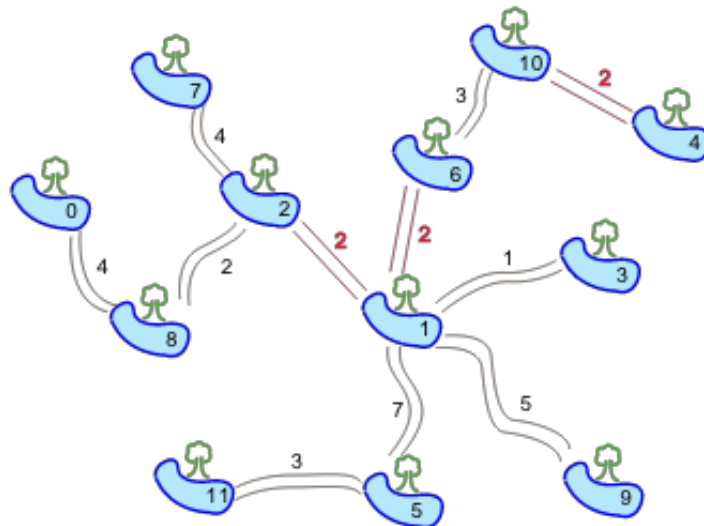
Inoltre, Canguro vuole far sì che Serpente possa spostarsi il più velocemente possibile. Canguro vuole quindi costruire i nuovi sentieri in modo che il massimo tempo di spostamento tra due billabong qualsiasi sia il più piccolo possibile. Aiuta Canguro e Serpente a determinare il massimo tempo di spostamento tra due billabong qualsiasi, una volta che Canguro ha costruito i nuovi sentieri in questo modo.

## Esempio



Nella figura soprastante ci sono  $N = 12$  billabong e  $M = 8$  sentieri. Sia  $L = 2$ , in modo che Serpente impieghi 2 giorni per percorrere ciascun nuovo sentiero. Canguro può costruire tre nuovi sentieri:

- tra i billabong 1 e 2;
- tra i billabong 1 e 6;
- tra i billabong 4 e 10.



La figura soprastante mostra l'insieme finale di sentieri. Il tempo massimo di spostamento è di 18 giorni, tra i billabong 0 e 11. Questo è il più piccolo risultato possibile: indipendentemente da come Canguro costruisca i sentieri, ci sono due billabong per i quali Serpente impiega almeno 18 giorni a spostarsi da uno all'altro.

---

## Implementazione

Devi sottoporre un file che implementa la funzione `travelTime()`, come segue:

### Funzione: `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;
```

### Descrizione

Questa funzione deve calcolare il massimo tempo di spostamento (misurato in giorni) tra ciascuna coppia di billabong, assumendo che Canguro abbia aggiunto  $N - M - 1$  sentieri in modo che tutti i billabong siano connessi e il massimo tempo di spostamento sia il più breve possibile.

## Parametri

- $N$ : Il numero di billabong.
- $M$ : Il numero di sentieri già esistenti.
- $L$ : I giorni che Serpente impiega ad attraversare i nuovi sentieri.
- $A$ ,  $B$  e  $T$ : Array di lunghezza  $M$  che specificano gli estremi e il tempo di spostamento per ciascun sentiero preesistente, così che l' $i$ -esimo sentiero colleghi i billabong  $A[i-1]$  e  $B[i-1]$ , e richieda  $T[i-1]$  giorni per spostarsi in entrambe le direzioni.
- *Restituisce*: Il massimo tempo di spostamento tra qualsiasi coppia di billabong, come descritto sopra.

---

## Sessione di esempio

La seguente sessione descrive l'esempio precedente:

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

---

## Limiti

- Tempo limite: 1 secondo
- Limite di memoria: 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	Punteggio	Limiti aggiuntivi dell'input
1	14	$M = N - 2$ , e ci sono esattamente uno o due sentieri preesistenti che partono da un billabong qualsiasi. In altre parole, ci sono due insiemi di billabong connessi, e in ciascun insieme i sentieri formano un percorso senza ramificazioni.
2	10	$M = N - 2$ e $N \leq 100$
3	23	$M = N - 2$
4	18	Esiste al massimo un sentiero preesistente che parte da un billabong qualsiasi.
5	12	$N \leq 3\,000$
6	23	(Nessuno)

## Testing

Il grader di esempio sul computer legge l'input dal file `dreaming.in`, che deve essere nel seguente formato:

- linea 1: `N M L`
- linee 2, ..., `M + 1`: `A[i] B[i] T[i]`

In particolare, l'esempio precedente viene codificato in questo modo:

```
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 relative al linguaggio

C/C++ Devi inserire `#include "dreaming.h"`.

Pascal Devi definire `unit Dreaming`. Tutti gli array sono numerati da `0` (non da `1`).

Vedi i template delle soluzioni sulla tua macchina per alcuni esempi.