



**International Olympiad in Informatics**  
**2013**  
 6-13 July 2013  
 Brisbane, Australia

**Мөрөөдөл**

Монгол — 1.0

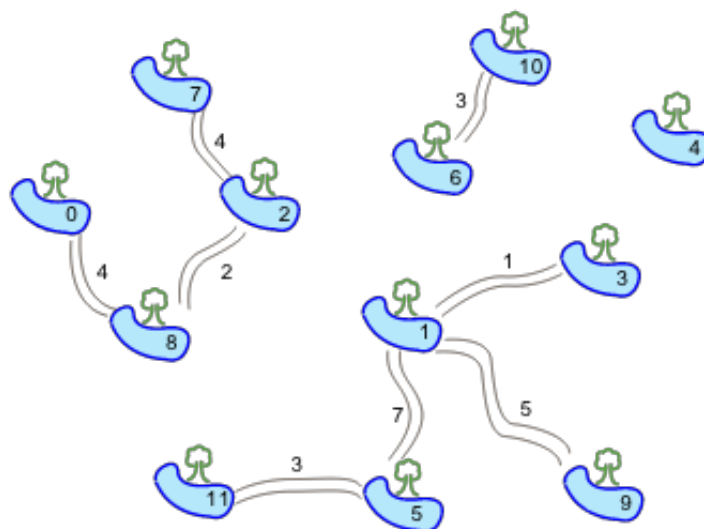
Энэ явдал маш эрт, ертөнц дөнгөж үүсээд, IOI-ийн тухай хэн ч мөрөөдөж байгаагүй үед болжээ.

Могой  $0, \dots, N-1$  тоонуудаар дугаарлагдсан билабонгтой (устай нүх) газар амьдардаг. Зарим хоёр билабонгийг холбосон, хоёр чиглэлд явж болох  $M$  тооны *замуудаар* Могой аялж чадна. Аль ч хоёр билабонг хоорондоо дээд тал нь нэг замуудын дарааллаар холбогдсон байх (шууд эсвэл шууд бусаар) боловч зарим нь хоорондоо огт холбогдоогүй байж болно(иймд,  $M \leq N-1$  байна). Могой ямар нэг замыг туулахдаа тодорхой тооны өдрийг зарцуулдаг ба тэдгээр нь зам бүрийн хувьд ялгаатай байж болно.

Могойнд Кенгуру гэдэг найз байдаг. Тэрээр Могойг аль ч хоёр билабонгийн хооронд аялж чаддаг байхаар  $N - M - 1$  тооны шинэ зам барихыг хүсч байгаа. Кенгуру аль ч хоёр билабонгийн хооронд зам барьж чадах ба Кенгуругийн барьсан нэг замаар Могой аялахдаа  $L$  өдөр зарцуулдаг.

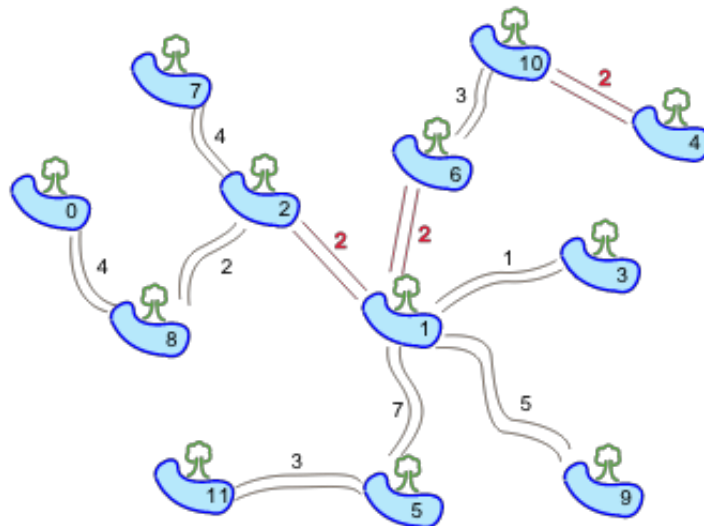
Мөн, Могойн аяллуудыг аль болох хурдан байлгахыг Кенгуру хүсч байгаа. Кенгуру шинэ замуудаа барихдаа бүх хос билабонгийн хооронд аялахад зарцуулах хугацаануудын хамгийн их утгыг аль болох бага байлгана. Ийм аргаар Кенгуру шинэ замуудаа барьсны дараа бүх хос билабонгийн хооронд аялахад зарцуулах хугацаануудын хамгийн их утгыг тодорхойлоход Кенгуру, Могой хоёрт тусална уу.

## Жишээ



Дээрх зурган дээр  $N = 12$  тооны билабонг болон  $M = 8$  тооны зам байна. Ямар ч шинэ замыг туулахад Могой 2 өдөр зарцуулах буюу  $L = 2$  байна гэж үз. Тэгвэл Кенгуру гурван шинэ зам барьж чадна:

- билабонг 1 болон 2-ын хооронд;
- билабонг 1 болон 6-ын хооронд;
- билабонг 4 болон 10-ын хооронд.



Дээрх зурган дээр замуудын эцсийн олонлогийг үзүүлэв. Хамгийн урт аялал нь билабонг 0 болон билабонг 11-ийн хооронд хийгдэх ба 18 өдрийг зарцуулна. Энэ бол боломжит хамгийн бага үр дүн юм. Учир нь Кенгуру замуудаа яаж ч барьсан аль нэг хоёр билабонгийн хооронд аялахад 18 эсвэл түүнээс олон өдрийг зарцуулах болно.

## Хэрэгжүүлэлт

Та доорх байдлаар `travelTime()` функцийг хэрэгжүүлэлтийг агуулсан файлыг илгээх ёстой:

### Таны Функц: `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;
```

Тодорхойлолт

Уг функц нь аль ч хоёр билабонгийн хооронд аялах хугацааны хамгийн их утгыг (өдрөөр хэмжигдэх) тооцоолох ёстой. Энд Кенгуру  $N - M - 1$  тооны замыг бүх билабонгууд хоорондоо холбогдсон байхаар болон уг хамгийн их утгыг аль болох бага байлгахаар нэмж барьсан гэж үзнэ.

## Параметрууд

- $N$ : Билабонгуудын тоо.
- $M$ : Өмнө нь байсан замуудын тоо.
- $L$ : Могойны нэг шинэ замыг туулахад зарцуулах хугацаа, өдрөөр.
- $A$ ,  $B$  болон  $T$ :  $i$ -р зам билабонг  $A[i-1]$  ба  $B[i-1]$ -г холбох бөгөөд аль ч чиглэлд туулахад  $T[i-1]$  өдөр зарцуулна гэдгийг илэрхийлэх өмнө нь байсан замуудын төгсгөлийн цэгүүд болон аяллын хугацааг тодорхойлох  $M$  урттай массивууд.
- *Буцаах утга*: Дээр өгүүлсэн аль ч хоёр билабонгийн хоорондох замуудыг туулах хугацаануудын хамгийн их утга.

## Жишээ Өгөгдөл

Доорх өгөгдөл нь дээрх жишээг тодорхойлно:

| 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                        |

## Хязгаарлалтууд

- Хугацааны хязгаарлалт: 1 секунд
- Санах ойн хязгаарлалт: 64 Мб
- $1 \leq N \leq 100,000$
- $0 \leq M \leq N - 1$
- $0 \leq A[i], B[i] \leq N - 1$
- $1 \leq T[i] \leq 10,000$

- $1 \leq L \leq 10,000$

## Дэд бодлогууд

| Дэд бодлого | Оноо | Оролтын Нэмэлт Хязгаарлалтууд                                                                                                                                                                                  |
|-------------|------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1           | 14   | $M = N - 2$ , мөн билабонг бүрээс яг нэг эсвэл хоёр өмнө нь байсан зам гарна. Өөрөөр хэлбэл, холбогдсон билабонгуудын хоёр олонлог байх ба олонлог бүрийн хувьд замууд нь салаалаагүй замын дарааллыг үүсгэнэ. |
| 2           | 10   | $M = N - 2$ ба $N \leq 100$                                                                                                                                                                                    |
| 3           | 23   | $M = N - 2$                                                                                                                                                                                                    |
| 4           | 18   | Билабонг бүрээс хамгийн ихдээ өмнө нь байсан нэг зам гарна.                                                                                                                                                    |
| 5           | 12   | $N \leq 3,000$                                                                                                                                                                                                 |
| 6           | 23   | (Байхгүй)                                                                                                                                                                                                      |

## Туршилт

Таны компьютер дээр байгаа жишээ шалгагч нь доорх форматтай `dreaming.in` файлаас оролтоо уншина:

- 1-р мөр: `N M L`
- 2, ...,  $M + 1$ -р мөрүүд: `A[i] B[i] T[i]`

Жишээ нь, дээрх жишээний өгөгдөл нь доор үзүүлсэн форматтай байх хэрэгтэй:

```
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
```

## Хэлнүүдийн тайлбар

C/C++ Энэ мөрийг агуулах ёстой `#include "dreaming.h"`.

Pascal Та `unit Dreaming`-г тодорхойлох ёстой. Бүх массивууд 0-ээс эхлэх ёстой (1-ээс эхлэхгүй).

Жишээ болгож компьютер дээрээ байгаа бодолтын загварыг харна уу.