



## International Olympiad in Informatics 2013

6-13 July 2013

Brisbane, Australia

**dreaming**

Armenian — 1.0

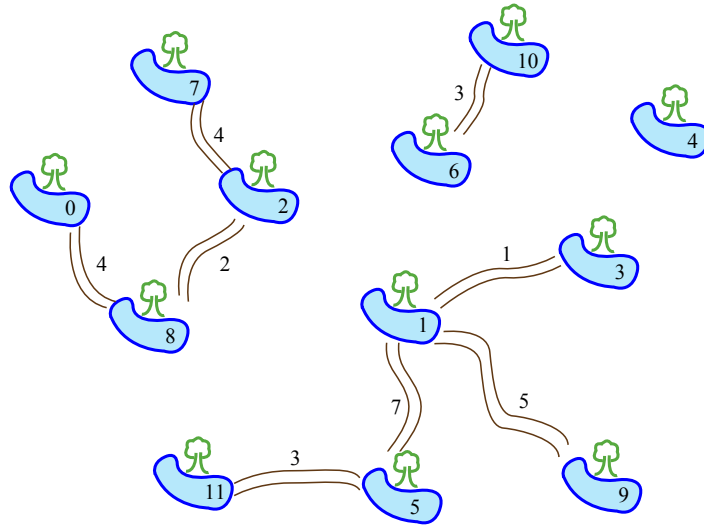
Այս պատմությունը եղել է վաղուց, երբ աշխարհը նոր էր, և IOI-ի մասին անգամ հնարավոր չէր երագել:

Օձն ապրում է մի երկրում, որտեղ կան "N" ջրափոսեր՝ համարակալված 0-ից N-1 թվերով: Կան M երկկողմանի արահետներ, որոնք իրար են կապում ջրափոսերի զույգեր, և օձը կարող է շրջել դրանցով: Ջրափոսերի յուրաքանչյուր զույգ կապակցված է (ուղղակիորեն կամ անուղղակիորեն) արահետների առավելագույնը մեկ հաջորդականությամբ, այսինքն ջրափոսերի որոշ զույգեր կարող են կապված չլինել (հետևաբար,  $M \leq N-1$ ): Յուրաքանչյուր արահետով անցնելու համար օձին անհրաժեշտ են որոշակի քանակությամբ օրեր և այդ թիվը տարբեր արահետների համար կարող է տարբեր լինել:

Օձի ընկեր Կենգուրուն ցանկանում է  $N - M - 1$  նոր արահետներ կառուցել, այնպես, որ Օձը կարողանա ցանկացած ջրափոսից ցանկացած ջրափոս գնալ: Կենգուրուն կարող է արահետ կառուցել ցանկացած երկու ջրափոսերի միջև, և Կենգուրուի պատրաստած յուրաքանչյուր արահետով անցնելու համար Օձին պետք է L օր:

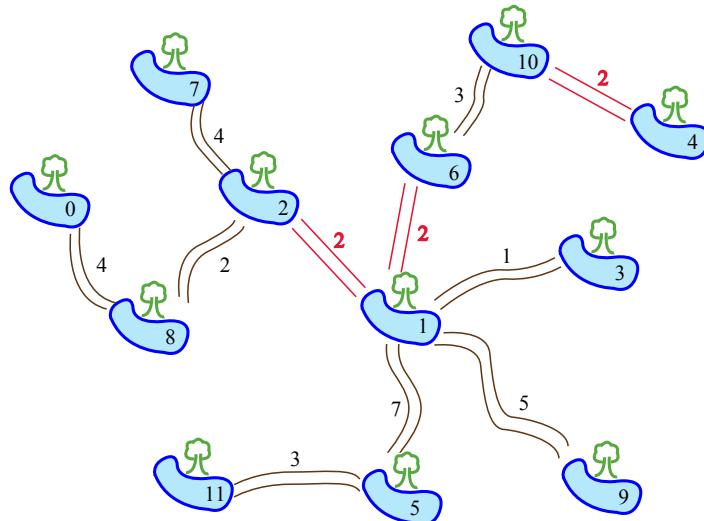
Բացի այդ, Կենգուրուն ցանկանում է Օձի ճամփորդությունները, որքան հնարավոր է, արագ դարձնել: Կենգուրուն պետք է նոր արահետներ կառուցի այնպես, որ ցանկացած երկու ջրափոսերի միջև ճամփորդելու ամենաերկար ժամանակը լինի փոքր որքան հնարավոր է: Օգնեք Կենգուրույին և Օձին. գտեք մի ջրափոսից մյուսը հասնելու ամենաերկար ժամանակը, երբ Կենգուրուն նոր արահետները այս կերպ կառուցած կլինի:

## Օրինակներ



Վերևի նկարում կա  $N = 12$  ջրափոս և  $M = 8$  արահետ: Դիցուք  $L = 2$ , այսինքն Օձը նոր արահետներից յուրաքանչյուրով կանցնի 2 օրում: Այդ դեպքում Կենգուրուն կարող է կառուցել երեք նոր արահետ.

- 1 և 2 ջրափոսերի միջև;
- 1 և 6 ջրափոսերի միջև;
- 4 և 10 ջրափոսերի միջև:



Վերևի նկարում պատկերված է արահետների վերջնական տեսքը: Ճամփորդելու ամենաերկար ժամանակը 18 օր է՝ 0 և 11 ջրափոսերի միջև: Դա հնարավոր ամենափոքր արդյունքն է: Կենգուրուն ինչպես էլ արահետներ կառուցի, կգտնվի ջրափոսերի որևէ զույգ, որոնցից մեկից մյուսը գնալու համար Օձին պետք է 18 օր կամ ավել ժամանակ:

## Իրականացումը

Դուք պետք է ուղարկեք ֆայլ, որտեղ իրականացված է `travelTime()` ֆունկցիան, as follows:

### Your Function: `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;
```

### Նկարագրությունը

Այս ֆունկցիան պետք է հաշվի ցանկացած երկու ջրափոսների միջև ճամփորդելու ամենաերկար ժամանակը (արտահայտված օրերով) այն ենթադրությամբ, որ Օձն ավելացրել է  $N - M - 1$  արահետ այնպես, որ բոլոր ջրափոսերը կապակցված են և ճամփորդելու ամենաերկար ժամանակը, որքան հնարավոր է, փոքր է:

### Պարամետրերը

- `N`: Ջրափոսերի քանակը:
- `M`: Արդեն գոյություն ունեցող ջրափոսերի քանակը:
- `L`: Օձը նոր արահետը քանի օրում է անցնում:
- `A`, `B` և `T`: Arrays of length `M` երկարության զանգվածներ, որոնք նկարագրում են նախապես գոյություն ունեցող արահետներից յուրաքանչյուրի ծայրակետերը և ճամփորդելու ժամանակը այնպես, որ  $i$  ընդ արահետը միացնում է `A[i-1]` և `B[i-1]` ջրափոսերը, և այդ արահետով անցնելու համար պետք է `T[i-1]` օր:
- *Returns*: Որևէ երկու ջրափոսերի միջև ճանապարհորդելու ամենաերկար ժամանակը, ինչպես նկարագրված է վերևում:

---

## Sample Session

The following session describes the example above:

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

## Constraints

- Time limit: 1 second
- Memory limit: 64 MiB
- $1 \leq N \leq 100,000$
- $0 \leq M \leq N - 1$
- $0 \leq A[i], B[i] \leq N - 1$
- $1 \leq T[i] \leq 10,000$
- $1 \leq L \leq 10,000$

## Ենթախնդիրներ

Subtask	Points	Լրացուցիչ մուտքային սահմանափակումներ
1	14	$M = N - 2$ , և յուրաքանչյուր ջրափոսից դուրս է գալիս ճիշտ մեկ կամ երկու արահետ: Այլ կերպ ասած, կա իրար հետ կապված ջրափոսերի երկու հավաքածու, որոնցից յուրաքանչյուրում նախապես տրված արահետները չճյուղավորվող ճանապարհ են կազմում:
2	10	$M = N - 2$ and $N \leq 100$
3	23	$M = N - 2$
5	12	$N \leq 3,000$

## Էքսպերիմենտ

Ձեր համակարգչում գրեյդերը կարդում է մուտքային տվյալները `dreaming.in` ֆայլից որը պետք է լինի հետևյալ ֆորմատով.

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

Օրինակ, վերևում նկարագրված դեպքը է տրված լինի հետևյալ կերպ.

```
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
```

## Դիտողություններ լեզուների վերաբերյալ

**C/C++** You must `#include "dreaming.h"`.

**Pascal** You must define the `unit Dreaming`. All arrays are numbered beginning at `0` (not `1`).

See the solution templates on your machine for examples.