



International Olympiad in Informatics 2013

6-13 July 2013

Brisbane, Australia

Giác mơ

Tiếng Việt – 1.0

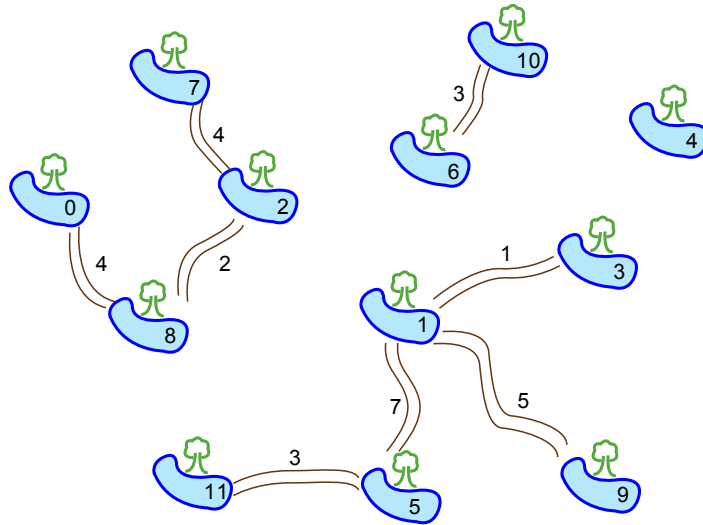
Câu chuyện này xảy ra từ cách đây rất lâu, khi trái đất mới hình thành, IOI thì còn chưa có trong mơ.

Serpent sống trong một vùng đất có N hồ nước, được đánh số $0, \dots, N-1$. Có M đường mòn hai chiều, nối các cặp hồ nước, mà theo đó Serpent có thể đi dạo. Mỗi cặp hồ nước được kết nối (trực tiếp hoặc gián tiếp) bởi nhiều nhất là một dãy các đường mòn, trong khi có thể có một số cặp hồ nước hoàn toàn không được kết nối (do đó, $M \leq N-1$). Mỗi đường mòn được gán cho một số xác định là số ngày cần thiết để Serpent đi dạo qua nó: con số này có thể khác nhau cho mỗi đường mòn.

Người bạn của Serpien là Kangaroo, muốn làm $N - M - 1$ con đường mòn mới, để giúp Serpien có thể đi dạo giữa mọi cặp hồ nước. Kangaroo có thể tạo ra đường mòn giữa bất kỳ cặp hồ nước nào, và mỗi đường mòn mà Kangaroo tạo ra đều đòi hỏi L ngày để Serpien đi dạo qua nó.

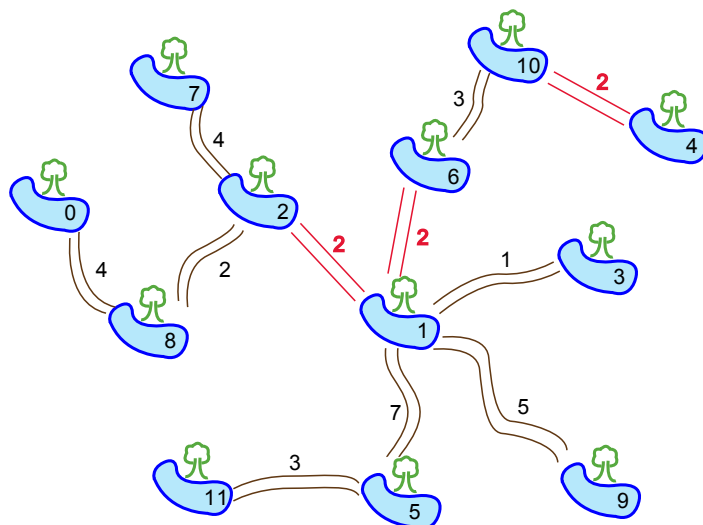
Ngoài ra, Kangaroo muốn làm cho hành trình của Serpent càng nhanh càng tốt. Kangaroo sẽ xây dựng những con đường mòn mới sao cho thời gian đi lại giữa hai hồ nước bất kỳ lớn nhất là càng nhỏ càng tốt. Giúp Kangaroo và Serpent xác định thời gian đi lại lớn nhất giữa bất kỳ hai hồ nước, sau khi Kangaroo đã xây dựng xong những con đường mòn mới theo yêu cầu nêu trên.

Các ví dụ



Trong hình vẽ ở trên có $N = 12$ hố nước và $M = 8$ đường mòn. Giả sử là $L = 2$, tức là mỗi đường mòn mới đòi hỏi Serpent mất 2 ngày để dạo qua. Khi đó Kangaroo sẽ xây dựng ba đường mòn mới:

- giữa các hố nước 1 và 2;
- giữa các hố nước 1 và 6;
- giữa các hố nước 4 và 10.



Hình vẽ ở trên cho thấy tập các đường mòn cuối cùng. Thời gian dạo lớn nhất là 18 ngày, giữa cặp hố nước 0 và 11. Đây là kết quả nhỏ nhất có thể — cho dù Kangaroo có xây dựng các đường mòn như thế nào chăng nữa, luôn tìm được cặp hố nước đòi hỏi thời gian mà Serpent đi lại giữa chúng là lớn hơn hoặc bằng 18 ngày.

Cài đặt

Bạn cần nộp một file cài đặt hàm `travelTime()`, như sau:

Hàm mà bạn phải xây dựng: `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;
```

Mô tả

Hàm này sẽ tính thời gian đi dạo lớn nhất (đo bởi số ngày) giữa mọi cặp hồ nước, với giả thiết là Kangaroo đã xây dựng thêm $N - M - 1$ đường mòn sao cho tất cả các hồ nước đều được kết nối và thời gian đi dạo lớn nhất này là càng nhỏ càng tốt.

Các tham số

- `N`: Số lượng hồ nước.
- `M`: Số lượng đường mòn đã có.
- `L`: Thời gian tính bằng ngày mà Serpent cần tốn để đi dạo theo đường mòn mới xây dựng.
- `A`, `B` và `T`: Các mảng độ dài `M` mô tả các điểm đầu nút và thời gian đi dạo của mỗi đường mòn đã tồn tại trước đó, trong đó đường mòn thứ `i` nối các hồ nước `A[i-1]` và `B[i-1]`, và đòi hỏi `T[i-1]` ngày để đi dạo qua nó theo một trong hai chiều.
- *Giá trị trả về*: Thời gian đi lại lớn nhất giữa mọi cặp hồ nước, như đã mô tả ở trên.

Phần ví dụ

Phần dưới đây mô tả ví dụ ở trên:

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

Các ràng buộc

- Giới hạn thời gian: 1 giây
- Giới hạn bộ nhớ: 64 MiB (megabyte)
- $1 \leq N \leq 100,000$
- $0 \leq M \leq N - 1$
- $0 \leq A[i], B[i] \leq N - 1$
- $1 \leq T[i] \leq 10,000$
- $1 \leq L \leq 10,000$

Subtasks

Subtask	Điểm	Hạn chế thêm về đầu vào
1	14	$M = N - 2$, và có đúng một hoặc hai đường mòn có từ trước đi ra từ mỗi hố nước. Nói cách khác, có hai tập gồm các hố nước được liên kết với nhau, và trong mỗi tập các con đường mòn tạo thành một đường đi không rẽ nhánh.
2	10	$M = N - 2$ và $N \leq 100$
3	23	$M = N - 2$
4	18	Có nhiều nhất một đường mòn có từ trước đi ra từ mỗi hố nước.
5	12	$N \leq 3,000$
6	23	(Không có)

Thực nghiệm

Chương trình chấm mẫu trên máy tính của bạn sẽ đọc dữ liệu vào từ file `dreaming.in`, được ghi theo khuôn dạng sau:

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

Chẳng hạn, ví dụ ở trên cần được ghi nhận theo khuôn dạng sau đây:

```
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
```

Chú ý về ngôn ngữ

C/C++ Bạn phải `#include "dreaming.h"`.

Pascal Bạn phải định nghĩa `unit Dreaming`. Tất cả các mảng được đánh chỉ số bắt đầu từ `0` (chứ không phải `1`).

Hãy xem mẫu các lời giải trên máy của bạn, như là những ví dụ.