



International Olympiad in Informatics 2013

6-13 July 2013
Brisbane, Australia

dreaming

Thai — 1.0

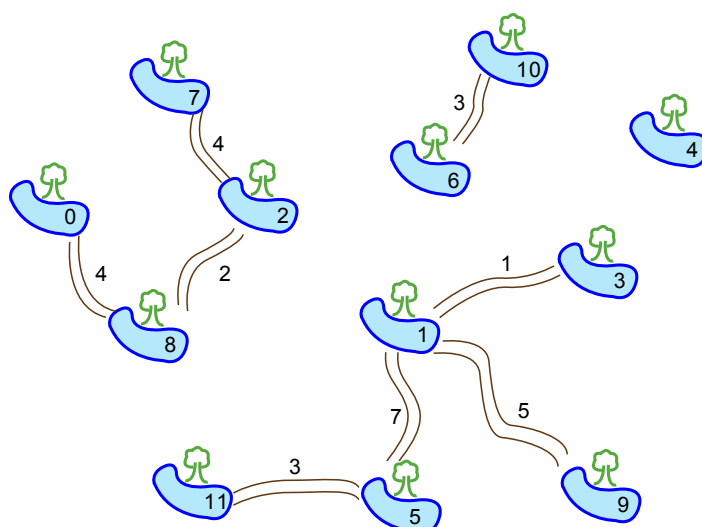
กาลครั้งหนึ่งนานมาแล้ว ตั้งแต่เมื่อโลกเพิ่งเกิดและยังไม่มีใครนึกถึง IOI

คุณงูอาศัยอยู่ในดินแดนที่มีบิลลาบอง (แอ่งน้ำขัง) N แอ่ง แต่ละแอ่งมีเลขกำกับตั้งแต่ $0, \dots, N-1$ และมีเส้นทาง (trail) M เส้นทางที่สามารถเดินทางไปและกลับได้ ซึ่งเชื่อมต่อแอ่งสองแอ่ง ทำให้คุณสามารถเดินทางระหว่างแอ่งสองแอ่งนี้ได้ แอ่งแต่ละแอ่งเชื่อมต่อกันด้วยลำดับของเส้นทางอย่างมากหนึ่งลำดับ (โดยทางตรง หรือทางอ้อม) แต่อาจมีแอ่งบางคู่ที่ไม่ได้เชื่อมกัน (นั่นคือ $M \leq N-1$) เส้นทางแต่ละเส้นทางต้องใช้เวลาในการเดินทางโดยจะระบุเวลาเป็นจำนวนวัน ซึ่งอาจแตกต่างกันไปตามเส้นทาง

เพื่อนของคุณงู ชื่อ คุณจิงโจ้ ต้องการจะสร้างเส้นทางใหม่จำนวน $N-M-1$ เส้นทาง ซึ่งจะทำให้คุณสามารถเดินทางระหว่างแอ่งแต่ละแอ่งได้ทุกคู่ คุณจิงโจ้สามารถสร้างเส้นทางระหว่างแอ่งคู่ใดๆได้ และแต่ละเส้นทางที่คุณจิงโจ้สร้างนั้น คุณงูจะใช้เวลาในการเดินทางเป็นเวลา L วัน

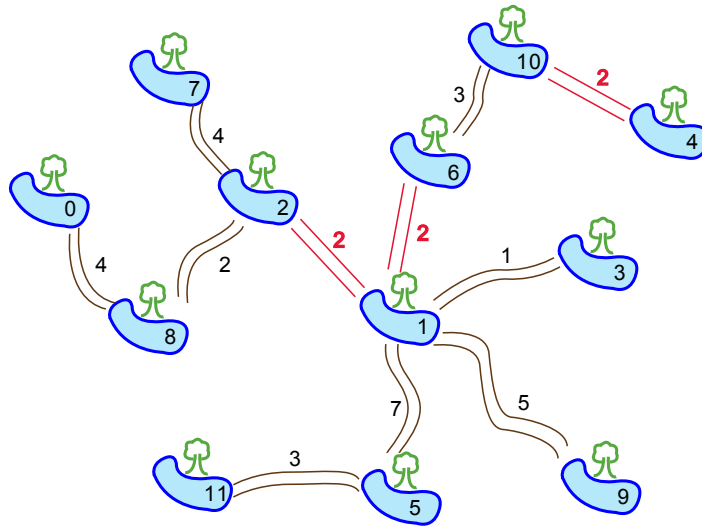
นอกจากนั้น คุณจิงโจ้ยังอยากให้คุณงูเดินทางได้เร็วที่สุด คุณจิงโจ้จึงจะสร้างเส้นทางใหม่ที่ทำให้เวลาในการเดินทางที่มากที่สุดระหว่างแอ่งคู่ใด ๆ มีค่าน้อยที่สุดเท่าที่เป็นไปได้ จงช่วยคุณจิงโจ้และคุณงูหาเวลาในการเดินทางที่มากที่สุดระหว่างแอ่งแต่ละแอ่งคู่ใดๆ หลังจากที่คุณจิงโจ้ได้สร้างเส้นทางใหม่ตามวิธีการนี้เรียบร้อยแล้ว

ตัวอย่าง



ในรูปด้านบน ประกอบด้วยแอ่ง $N = 12$ แอ่ง และเส้นทาง $M = 8$ เส้นทาง สมมติให้ $L = 2$ ดังนั้นคุณงูจะเดินทางบนเส้นทางที่สร้างขึ้นใหม่โดยใช้เวลา 2 วัน คุณจิงโจ้สามารถสร้างเส้นทางใหม่ 3 เส้นทางดังนี้

- ระหว่างแ่งที่ 1 และ 2
- ระหว่างแ่งที่ 1 และ 6
- ระหว่างแ่งที่ 4 และ 10



รูปด้านบนแสดงผลลัพธ์สุดท้ายของเส้นทางทั้งหมด เส้นทางที่ใช้เวลาในการเดินทางมากที่สุด คือ เส้นทางระหว่างแ่งที่ 0 และแ่งที่ 11 ซึ่งใช้เวลาในการเดินทาง 18 วัน นี่เป็นเวลาน้อยที่สุดที่เป็นได้ เพราะไม่ว่าคุณจึงใจจะสร้างเส้นทางอย่างไร จะมีแ่งบางคู่ที่ทำให้คุณงต้องเดินทาง 18 วันหรือมากกว่านั้นเสมอ

การเขียนโปรแกรม

คุณควรส่งไฟล์ที่เขียนฟังก์ชัน `travelTime()` ดังต่อไปนี้

ฟังก์ชัน `travelTime()` ของคุณ

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;
```

คำอธิบาย

ฟังก์ชันนี้ทำหน้าที่คำนวณเวลาในการเดินทางที่มากที่สุด (หน่วยเป็นวัน) ระหว่างแ่งคู่ใด ๆ โดยมีสมมติฐานว่าคุณจึงใจได้เพิ่มเส้นทางจำนวน $N-M-1$ เส้นทาง ที่ทำให้แ่งเชื่อมต่อกันทั้งหมดได้ และเวลาในการเดินทางที่มากที่สุด มีค่าน้อยที่สุดเท่าที่เป็นไปได้

พารามิเตอร์

- `N`: จำนวนแ่งน้ำขัง

- M : จำนวนเส้นทางที่มีอยู่แล้ว
- L : จำนวนวันที่คุณใช้ในการเดินทางบนเส้นทางสายใหม่
- A , B และ T : อาเรย์ที่มีความยาว M ที่ระบุจุดปลายทั้งสองของเส้นทางและเวลาในการเดินทางระหว่างเส้นทางที่มีอยู่แล้ว กล่าวคือ เส้นทางที่ i เชื่อมระหว่างแอง $A[i-1]$ และแอง $B[i-1]$ และใช้เวลา $T[i-1]$ ในการเดินทางไปหรือกลับ
- *คืนค่า*: เวลาในการเดินทางที่มากที่สุดระหว่างแองคู่ใด ๆ ตามที่ได้อธิบายด้านบน

ตัวอย่างการติดต่อ

การติดต่อต่อไปนี้อธิบายตัวอย่างด้านบน

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

เงื่อนไขบังคับ

- จำกัดเวลา 1 วินาที
- จำกัดหน่วยความจำ 64 MiB
- $1 \leq N \leq 100,000$
- $0 \leq M \leq N - 1$
- $0 \leq A[i], B[i] \leq N - 1$
- $1 \leq T[i] \leq 10,000$
- $1 \leq L \leq 10,000$

ปัญหาย่อย

ปัญหาย่อย	คะแนน	ข้อจำกัดเพิ่มเติมของข้อมูลนำเข้า
1	14	$M = N - 2$ และมีแค่หนึ่งหรือสองเส้นทางที่มีอยู่แล้ว เชื่อมกับแ่งแต่ละแ่ง หรือพูดอีกอย่างหนึ่งว่า มีแ่งนำเข้าที่เชื่อมต่อกันสองเซต และในแต่ละเซตประกอบไปด้วยเส้นทางที่เชื่อมต่อกันเป็น path ที่ไม่มีการแยกสาขาออก
2	10	$M = N - 2$ และ $N \leq 100$
3	23	$M = N - 2$
4	18	มีเส้นทางที่มีอยู่แล้วอย่างมากหนึ่งเส้นที่เชื่อมกับแ่งแต่ละแ่ง
5	12	$N \leq 3,000$
6	23	(ไม่มี)

การทดลอง

เกรดเตอร์ตัวอย่างบนเครื่องคอมพิวเตอร์ของคุณจะอ่านข้อมูลนำเข้าจากไฟล์ชื่อ `dreaming.in` ซึ่งต้องมีรูปแบบดังต่อไปนี้

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

ตัวอย่างเช่น ตัวอย่างด้านบนควรอยู่ในรูปแบบดังต่อไปนี้

```
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
```

หมายเหตุของภาษา

C/C++ คุณจะต้องระบุ `#include "dreaming.h"` ที่ส่วนหัวของโปรแกรม
Pascal คุณจะต้องนิยาม `unit Dreaming` อาเรย์ทั้งหมดจะเริ่มนับที่ 0 (ไม่ใช่ 1).

คุณสามารถดูตัวอย่างได้จากเทมเพลตในเครื่องของคุณ