



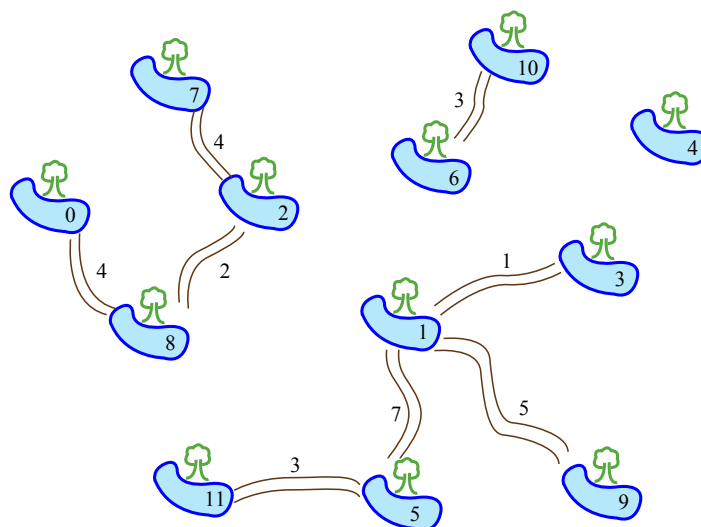
이 이야기는 세상의 태초에, IOI라는 것은 꿈꿀 수도 없었던 아주 먼 옛날에 생긴 일이다.

$0, \dots, N-1$ 까지 번호가 붙여진 N 개의 빌라봉(호수)이 있는 땅에 큰 뱀이 살고 있다. 이 땅에는 M 개의 길이 있고, 각각의 길은 한 쌍의 빌라봉을 연결하며, 이 길을 통해 뱀은 양방향으로 이동할 수 있다. 임의의 빌라봉 쌍은 길을 따라서 (직접 또는 간접적으로) 최대 하나의 경로로 연결이 되어 있다. 단, 어떤 빌라봉 쌍들은 연결되어 있지 않을 수도 있다 (즉, $M \leq N-1$ 이다). 뱀이 하나의 길을 지나가는 데에는 몇 일의 시간이 걸리는데, 이 시간은 길마다 다를 수 있다.

뱀의 친구 캥거루는 $N-M-1$ 개의 새로운 길을 만들어 뱀이 모든 빌라봉 쌍 사이를 오갈 수 있도록 하고 싶다. 캥거루는 임의의 두 빌라봉을 선택하여 그 사이에 길을 만들 수 있으며, 캥거루가 만든 길을 통행하는 데에는 항상 L 일이 걸린다.

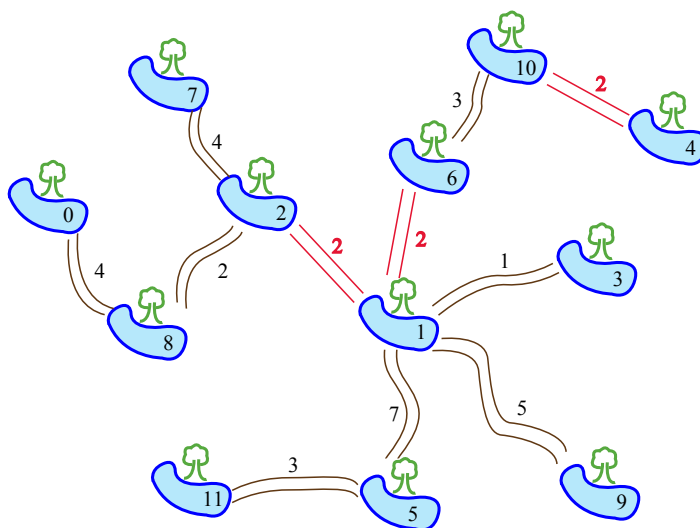
캥거루는 또한 뱀이 최대한 빨리 이동할 수 있기를 원한다. 캥거루는 임의의 두 빌라봉 사이를 오가는데 드는 최대 시간이 최소가 되도록 길을 만들 것이다. 캥거루가 이렇게 길을 만든 뒤 뱀이 두 빌라봉 사이를 오가는데 드는 최대 시간을 계산하시오.

예시



위 그림에는 $N=12$ 개의 빌라봉과 $M=8$ 개의 길이 있다. 새로 만들어지는 길을 뱀이 통행하는 데에는 2일이 걸린다고 가정하자 (즉, $L=2$ 이다). 그러면 캥거루는 아래와 같이 세 개의 길을 만들 수 있다.

- 빌라봉 1과 2 사이,
- 빌라봉 1과 6 사이,
- 빌라봉 4와 10 사이.



모든 길이 만들어진 뒤의 모습이 위 그림에 나타나 있다. 여기서 두 빌라봉 사이의 최대 통행 시간은 18일인데 (빌라봉 0과 11 사이), 이것이 가능한 가장 작은 값이다. 즉, 캥거루가 어떤 식으로 새로운 길을 만들더라도, 뱀이 통행하는 데 18일 이상이 걸리는 빌라봉 쌍이 항상 존재한다.

구현

다음 조건을 만족하는 함수 `travelTime()` 을 구현한 파일을 제출하시오.

구현해야 하는 함수: **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;
```

설명

이 함수는 모든 빌라봉이 연결되고 최대 통행 시간이 최소가 되도록 $N-M-1$ 개의 길을 만든 뒤의 최대 통행 시간(일 단위)을 계산하여야 한다.

파라미터

- N : 빌라봉의 개수.
- M : 이미 존재하는 길들의 개수.
- L : 뱀이 새로 지어진 길을 통행하는데 걸리는 시간 (일 단위).
- A, B, T : 이미 존재하는 길들이 연결하는 빌라봉들과 각각을 통행하는데 걸리는 시간을 나타내는 길이가 M 인 배열들. i 번째 길은 빌라봉 $A[i-1]$ 과 $B[i-1]$ 을 이으며, 통행 시간은 양방향 모두 $T[i-1]$ 일이다.
- 리턴값: 위에서 설명했던 것과 같이, 임의의 한 쌍의 빌라봉을 통행하는데 걸리는 최대 시간.

예제 세션

다음 예제 세션은 위의 예시를 나타낸 것이다.

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

제약 조건

- 시간 제한: 1초
- 메모리 제한: 64 MB
- $1 \leq N \leq 100,000$
- $0 \leq M \leq N - 1$
- $0 \leq A[i], B[i] \leq N - 1$
- $1 \leq T[i] \leq 10,000$
- $1 \leq L \leq 10,000$

서브태스크

서브태스크	배점	추가적인 입력 제한 사항
1	14	$M = N - 2$ 이며, 각 빌라봉은 정확히 한 개 또는 두 개의 길들과 연결이 되어 있다. 다시 말해서, 연결된 빌라봉 집합이 두 개가 있고, 각 집합은 갈림길이 없는 하나의 경로로 되어 있다.
2	10	$M = N - 2, N \leq 100$
3	23	$M = N - 2$
4	18	각 빌라봉은 최대 한 개의 길과 연결이 되어 있다.
5	12	$N \leq 3,000$
6	23	(없음)

테스트용 입력 형식

당신의 컴퓨터에 있는 샘플 그레이더는 `dreaming.in` 으로부터 입력을 받아들이며, 입력 형식은 아래와 같아야 한다.

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

예를 들어, 위의 예시는 아래와 같이 입력되어야 한다.

```
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
```

언어 유의사항

- C/C++ 당신의 프로그램은 `#include "dreaming.h"` 명령어를 통해 헤더 파일을 추가시켜야 한다.
- Pascal 당신의 프로그램은 `unit Dreaming` 을 정의해야 한다. 모든 배열의 인덱스는 1 이 아닌 0 부터 시작한다.

예시를 위해서 컴퓨터에 있는 솔루션 템플릿을 참조하십시오.