

기지 간소화 (base)

먼 미래, 인류는 수많은 외계 행성들에 진출하였다. 행성 X도 그 중 하나로, 우주 탐사 기업 MR은 행성 X에 기지들을 지어 탐사 및 자원 채취 활동을 수행하고 있었다.

행성 X에는 N 개의 기지와 기지들을 잇는 $N - 1$ 개의 양방향 통로가 있으며, 임의의 서로 다른 두 기지를 통로만을 사용하여 오갈 수 있다. 즉, 행성 X의 기지와 통로는 트리 구조를 이룬다.

기지에는 각각 0 이상 $N - 1$ 이하의 서로 다른 번호가 붙어 있다. 또 모든 $0 \leq i \leq N - 2$ 에 대해서 i 번 도로는 $U[i]$ 번 도시와 $V[i]$ 번 도시를 연결하며 통로의 길이는 $W[i]$ km이다.

어느덧 행성 X의 개발도 안정기에 접어들었다. 모든 기지와 통로를 유지하는 것은 비용이 많이 들기 때문에, MR에서는 일부 기지들만 남기고 나머지를 비활성화하기로 하였다.

어떤 (s, e) ($0 \leq s \leq e \leq N - 1$)에 대해 $s, s + 1, \dots, e$ 번 기지만 남기기로 했다고 하자. 이 때 유지 비용은 다음과 같이 정의된다.

- 0개 이상의 통로를 골라 다음 조건을 만족시키자. 이 때, 고른 통로들의 길이의 합이 최소가 되도록 고른다. (통로를 0개 고른 경우, 길이의 합은 0 km이다.)
 - 임의의 u, v ($s \leq u < v \leq e$)에 대해, u 번 기지와 v 번 기지를 고른 통로들만 이용해서 서로 오고갈 수 있다. 중간에 비활성화된 기지를 거치는 것은 상관 없다.
- 고른 통로들의 길이의 합이 C km일 때, 유지 비용은 C 이다.

어떤 기지를 남길지는 아직 정하지 않았기 때문에, MR은 일단 가능한 모든 (i, j) ($0 \leq i \leq j \leq N - 1$) 쌍에 대해 $i, i + 1, \dots, j$ 번 기지만 남겼을 때의 유지 비용을 모두 합한 값을 알고 싶어 한다. 여러분은 MR을 위해 이 값을 구해야 한다. 단, 값이 매우 커질 수 있으므로 소수 1000 000 007로 나눈 나머지를 구해야 한다.

함수 목록 및 정의

여러분은 아래 함수를 구현해야 한다.

```
int maintenance_costs_sum(vector<int> U, vector<int> V, vector<int> W)
```

- 이 함수는 단 한 번만 호출된다.
- U, V, W : 크기가 $N - 1$ 인 정수 배열. 모든 i ($0 \leq i \leq N - 2$)에 대해, $U[i]$ 번 기지와 $V[i]$ 번 기지를 잇는 길이 $W[i]$ km의 통로가 있다.
- 이 함수는 가능한 모든 (i, j) ($0 \leq i \leq j \leq N - 1$) 쌍에 대해 $i, i + 1, \dots, j$ 번 기지만 남겼을 때의 유지 비용을 모두 합한 값을 1000 000 007로 나눈 나머지를 반환해야 한다.

제출하는 소스 코드의 어느 부분에서도 입출력 함수를 실행해서는 안 된다.

제약 조건

- $2 \leq N \leq 250\,000$
- 모든 i 에 대해 $0 \leq U[i], V[i] \leq N - 1; U[i] \neq V[i]$ ($0 \leq i \leq N - 2$)
- 모든 i 에 대해 $1 \leq W[i] \leq 10^9$ ($0 \leq i \leq N - 2$)

부분문제

- (5점)
 - $N \leq 300$
- (6점)
 - $N \leq 4\,000$
- (10점)
 - 기지에 매겨진 번호는 0번 기지를 루트로 한 트리의 전위 순회 순서 (Preorder) 중 하나이다.
- (26점)
 - 각 기지에 연결된 통로가 최대 2개이다.
- (53점)
 - 추가적인 제약 조건이 없다.

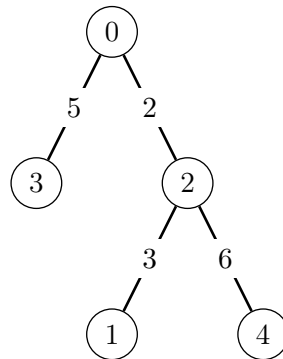
예제 1

$N = 5, U = [0, 2, 2, 0], V = [2, 1, 4, 3], W = [2, 3, 6, 5]$ 인 경우를 생각해 보자.

그레이더는 다음 함수를 호출한다.

```
maintenance_costs_sum([0, 2, 2, 0], [2, 1, 4, 3], [2, 3, 6, 5])
```

아래 그림은 행성 X의 기지 및 통로의 배치를 나타낸다.



모든 가능한 (i, j) 쌍에 대해 유지 비용을 각각 구해 보면 아래의 표와 같다.

$i \backslash j$	0	1	2	3	4
0	0	5	5	10	16
1	-	0	3	10	16
2	-	-	0	7	13
3	-	-	-	0	13
4	-	-	-	-	0

함수는 98을 반환해야 한다.

Sample grader

Sample grader는 아래와 같은 형식으로 입력을 받는다.

- Line 1: N
- Line $2 + i$ ($0 \leq i \leq N - 2$): $U[i] V[i] W[i]$

Sample grader는 다음을 출력한다.

- Line 1: `maintenance_costs_sum`이 반환한 값

Sample grader는 실제 채점에서 사용하는 그레이더와 다를 수 있음에 유의하라.