

사이버랜드

사이버랜드가 APIO 3742를 주최하기로 했다. 이 세계에서는, 0부터 $N - 1$ 까지 정수로 표현되는 N 개의 나라가 있고, 0부터 $M - 1$ 까지 정수로 표현되는 M 개의 양방향 도로가 있다. i 번 도로는 ($0 \leq i < M$) 서로 다른 두 나라 $x[i]$ 와 $y[i]$ 를 연결하며, 이 도로를 통과하는데 $c[i]$ 만큼의 시간이 필요하다. 당신은 0번 나라에 살고 있는데, 다른 모든 APIO 참가국은 이미 사이버랜드에 도착해 있다. 사이버랜드는 H 번 나라이다. 당신 나라에서 사이버랜드까지 최소한의 시간으로 가는 경로를 찾는데 당신의 도움이 필요하다.

방문하는 나라들 중에는 특별한 능력을 가진 나라들이 있다. 어떤 나라들은 지금까지 여행에 쓴 시간을 0으로 바꾸어줄 수 있다. 또 어떤 나라들은 지금까지 여행에 쓴 시간을 절반으로 줄여줄 수 있다. 이미 방문한 나라를 또 방문할 수 있다. 매번 어떤 나라를 방문할 때마다, 이 나라가 갖고 있는 특별한 능력을 쓸지 말지를 고를 수 있다. 특별한 능력을 가진 나라를 한번 방문하여 떠나기 전까지 이 나라가 가진 능력을 최대 한 번 쓸 수 있다. (즉, 이 능력을 여러번 쓰려면 이 나라를 여러번 방문해야 한다.) 또, 시간을 절반으로 줄여줄 수 있는 능력은 최대 K 번 쓸 수 있다. 일단 사이버랜드에 도착하면 APIO를 봐야 하기 때문에 더 이상 움직일 수 없다.

배열 arr 가 주어지는데, arr_i ($0 \leq i < N$)는 i 번 나라가 가진 특별한 능력을 나타낸다. 특별한 능력은 3가지 값으로 표현한다.

- $arr_i = 0$ 이면 이 나라는 여행에 쓴 시간을 0으로 바꾸어줄 수 있다.
- $arr_i = 1$ 이면 이 나라는 특별한 능력이 없어서 여행 시간을 바꾸어 줄 수 없다.
- $arr_i = 2$ 이면 이 나라는 여행에 쓴 시간을 절반으로 줄여줄 수 있다.

$arr_0 = arr_H = 1$ 임이 보장된다. 즉, 당신 나라와 사이버랜드는 특별한 능력이 없다.

당신 나라는 가장 빨리 사이버랜드에 도착해서 APIO를 보고 싶다. 만약 사이버랜드에 가는 경로가 없다면, 답은 -1 이어야 한다.

상세 구현

다음 함수를 구현해야 한다.

```
double solve(int N, int M, int K, int H, std::vector<int> x, std::vector<int> y, std::vector<int> c, std::vector<int> arr);
```

- N : 나라의 수.
- M : 양방향 도로의 수.
- K : 시간을 절반으로 줄여줄 수 있는 능력을 쓸 수 있는 횟수의 상한
- H : 사이버랜드를 나타내는 수.

- x, y, c : 길이 M 인 세 배열. 순서쌍 $(x[i], y[i], c[i])$ 은 i 번 양방향 도로가 $x[i]$ 와 $y[i]$ 를 연결하며 통과 하는데 걸리는 시간이 $c[i]$ 라는 것을 나타낸다.
- arr : 길이 N 인 배열로, $arr[i]$ 는 i 번 나라의 특별한 능력을 나타낸다.
- 이 함수의 리턴값은 당신 나라에서 사이버랜드까지 가는데 걸리는 최소 시간이다. 만약 사이버랜드 까지 도달할 수 없다면 -1 을 리턴한다.
- 이 함수는 한 번보다 많이 호출될 수 있다..

여러분의 답이 ans_1 이고 정답이 ans_2 이라면, $\frac{|ans_1 - ans_2|}{\max\{ans_2, 1\}} \leq 10^{-6}$ 이면 정답으로 간주된다.

Note: 이 함수는 한번보다 많이 호출될 수 있기 때문에, 이전 호출 결과로 남아있는 데이터가 다음 호출에 영향을 미칠 수 있다는데 유의하라.

예제

예제 1

다음 함수 호출을 생각해보자.

```
solve(3, 2, 30, 2, {1, 2}, {2, 0}, {12, 4}, {1, 2, 1});
```

사이버랜드까지 가는 유일한 경로는 $0 \rightarrow 2$ 인데, 사이버랜드에 일단 도착하면 더 이상 움직일 수 없기 때문이다. 시간은 다음과 같다.

나라 번호	통과 시간
0	0
2	$0 + 4 \rightarrow 4$ (합) $\rightarrow 4$ (특별 능력)

따라서, 함수의 리턴값은 4여야 한다.

예제 2

다음 함수 호출을 생각해보자.

```
solve(4, 4, 30, 3, {0, 0, 1, 2}, {1, 2, 3, 3}, {5, 4, 2, 4}, {1, 0, 2, 1});
```

당신 나라에서 사이버랜드까지 가는 경로는 둘 있다. $0 \rightarrow 1 \rightarrow 3$ 과 $0 \rightarrow 2 \rightarrow 3$ 이다.

$0 \rightarrow 1 \rightarrow 3$ 를 따라 간다면, 통과 시간은 다음과 같다.

나라 번호	통과 시간
0	0
1	$0 + 5 \rightarrow 5$ (합) $\rightarrow 0$ (특별 능력)
3	$0 + 2 \rightarrow 2$ (합) $\rightarrow 2$ (특별 능력)

$0 \rightarrow 2 \rightarrow 3$ 를 따라 간다면, 통과 시간은 다음과 같다.

나라 번호	통과 시간
0	0
2	$0 + 4 \rightarrow 4$ (합) $\rightarrow 2$ (특별 능력)
3	$2 + 4 \rightarrow 6$ (합) $\rightarrow 6$ (특별 능력)

따라서, 함수의 리턴값은 2여야 한다.

제약 조건

- $2 \leq N \leq 10^5, \sum N \leq 10^5$.
- $0 \leq M \leq \min\{10^5, \frac{N(N-1)}{2}\}, \sum M \leq 10^5$.
- $1 \leq K \leq 10^6$.
- $1 \leq H < N$.
- $0 \leq x[i], y[i] < N, x[i] \neq y[i]$.
- $1 \leq c[i] \leq 10^9$.
- $arr[i] \in \{0, 1, 2\}$.
- 어떤 두 나라를 고르더라도 이 둘을 연결하는 도로는 최대 한 개임이 보장된다.

부분 문제

1. (5 점): $N \leq 3, K \leq 30$.
2. (8 점): $M = N - 1, K \leq 30, arr[i] = 1$, 어느 두 나라를 고르더라도 둘 사이를 연결하는 경로가 있다.
3. (13 점): $M = N - 1, K \leq 30, arr[i] \in \{0, 1\}$, 어느 두 나라를 고르더라도 둘 사이를 연결하는 경로가 있다.
4. (19 점): $M = N - 1, K \leq 30, x[i] = i, y[i] = i + 1$.
5. (7 점): $K \leq 30, arr[i] = 1$.
6. (16 점): $K \leq 30, arr[i] \in \{0, 1\}$.
7. (29 점): $K \leq 30$.
8. (3 점): 추가적인 제약 조건이 없다.

샘플 그레이더

샘플 그레이더는 다음 양식으로 입력을 읽는다.

- line 1: T

T 개의 테스트 케이스는 각각 다음과 같이 구성되어 있다.

- line 1: $N \ M \ K$
- line 2: H
- line 3: $arr[0] \ arr[1] \ arr[2] \ \dots \ arr[N - 1]$

- line $4 + i$ ($0 \leq i \leq M - 1$): $x[i] \ y[i] \ z[i]$

샘플 그레이더는 다음 양식으로 답을 출력한다.

각각의 테스트 케이스마다 :

- line 1: `solve`의 리턴값