

C2. Two Trains

[Problem Analysis]

도시들과 도시 사이의 기차 노선을 graph로 표현할 때, graph $G = (V, E)$ 에 대하여 $V = \{v_1, v_2, \dots, v_n\}$ 는 도시(vertex), E 는 두 도시를 잇는 기차의 선로(edge)를 가리킨다.

이 때, α 와 β 가 각각 v_1, v_2 에서 v_n 까지 도달하는데에 총 사용되는 비용인 total fuel의 가장 작은 값을 구해야 한다. shortest path일 때의 distance라고 봐도 되겠다.

α 와 β 는 같은 도시에서 coupling할 수 있으므로, 만약 두 기차의 path가 겹친다면, coupling 후 이동하는 것이 더 적은 fuel이 부과된다($r < p+q$). 단, coupling후에는 decoupling이 불가하다는 것에 유념해야한다.

따라서 결국 구하고자 하는 것은 α 와 β 가 v_n 까지 도달하는데에 가장 작게 소모하는 fuel cost인데, coupling을 어느 도시에서 하여 이동하는 것이 이 연료의 소모량을 최소로 하느냐를 알아야한다.

만약 coupled point를 $v_i \in V$ 라고 하면, v_i 에서 두 기차가 coupling되어 함께 이동하는 minimum fuel cost(가장 적은 연료를 쓰는 경우)는 각 edge들의 fuel을 r 로 하는 v_i 부터 v_n 까지의 minimum fuel cost ($\alpha + \beta$ fuel)과 같다.

또한, 해당 coupling point에 도달하기까지 α 와 β 의 각 minimum total fuel은 아래와 같다.

1. α : 각 edge의 fuel을 p 로 하는 v_1 부터 v_i 까지의 minimum fuel cost (α fuel)
2. β : 각 edge의 fuel을 q 로 하는 v_2 부터 v_i 까지의 minimum fuel cost (β fuel)

따라서 중요한 구하고자 하는 최종적인 minimum total fuel은 특정 coupling point인 v_i 로부터 파생되는 (α fuel + β fuel + ($\alpha + \beta$) fuel)들 중 가장 최소의 값을 찾으면 된다.

[Solution]

1. p, q, r, N, M 을 받고, 각 도시간의 노선인 edge를 M 만큼 받아서 G 를 adjacency list의 형태로 초기화한다.
2. α 가 출발하는 v_1 을 시작점으로 두고, p 를 weight으로 하여 모든 V 에 대한 minimum fuel cost를 저장하는 배열 $fuel_alpha$ 를 만든다.
3. β 가 출발하는 v_2 를 시작점으로 두고, q 를 weight으로 하여 모든 V 에 대한 minimum fuel cost를 저장하는 배열 $fuel_beta$ 를 만든다.
4. α, β 가 도착하는 v_n 를 시작점으로 두고, r 를 weight으로 하여 모든 V 에 대한 minimum fuel cost를 저장하는 배열 $fuel_coupled$ 을 만든다.
5. $fuel_alpha, fuel_beta, fuel_coupled$ (이하 $fuel$)을 구하기 위해 G 와 각 weight(fuel consumption), 그리고 출발 도시인 $start$ 를 사용한다.
 - a. 배열 $fuel$ 에 있어서 모든 fuel cost를 INF로 초기화한다. 단, 출발 도시의 fuel은 0으로 초기화한다.
 - b. 출발 도시인 v_i 를 fuel cost가 작은 순으로 우선순위를 부여할 수 있는 컨테이너에 삽입한다.
 - i. 컨테이너에서 v_i 를 꺼내고, 해당 v_i 에 연결되어있는 v_j 들을 조회한다.
 - ii. 해당 도시(v_j)로 이동하는데에 드는 fuel cost가 기존 fuel 배열에 저장되어있는 fuel cost보다 작다면 fuel cost를 v_j 의 fuel cost + 해당 경우의 fuel consumption(p, q, r)로 새로 업데이트해준다. 만약 업데이트가 된다면 컨테이너에 v_j 를 삽입한다.
 - c. 컨테이너에 남아있는 도시가 없을 때까지 b를 반복한다.
6. G 의 모든 $v_i \in V$ 에 대하여, coupling하는 모든 경우에 생기는 minimum fuel cost를 구한다.
 - a. $fuel_alpha[i]$ 와 $fuel_beta[i]$, $fuel_coupled[i]$ 를 더한다.
 - b. $i = 1$ 부터 N 까지 반복하여, 그 중 최소값을 구한다.
7. 해당 최소값을 출력한다.

[Time Complexity] \

연결된 도시의 pair 수 M 에 대하여, 5-b 과정 중 인접 도시를 조회하는데에 $O(M)$ 그리고 조회된 도시를 우선순위 정렬 컨테이너에 삽입할 때 $O(\log M)$ 이 소모되므로, Time complexity는 **$O(M \log M)$** 이다.