

# “고등부 4번. 점수 경주” 문제 풀이

작성자: 이종영

## 부분문제 1

어떤 정점  $v$ 를 지나가는 모든 참가자들은  $v$ 까지 같은 전략을 취함을 이용하자. 최적 전략을 DFS로 구현하면서 현재  $S$ 와  $C$ 를 관리하면  $O(N^2)$ 에 문제를 해결할 수 있다.

## 부분문제 5

각 정점  $u$ 에 대해,  $v \leq u$ 인  $v$ 들을 처리하자.  $v > u$ 인  $v$ 들은 뒤집은 후 같은 방법으로 처리할 수 있다.

1번 정점에서  $u$ 까지의 거리를  $D_u$ 라고 하자.  $D_u - D_v < 0$ 인 마지막  $v$ 에 대해,  $v$ 를 지나는 참가자들은  $v$ 에서 점수를 0으로 만든다. 또한  $v$ 를 지나기 전에는 점수를 0으로 만들지 않는다. 이를 이용하면  $v$ 의 답에서  $u$ 의 답을 얻을 수 있으며, 동적 프로그래밍을 통해 답을 구할 수 있다.  $v$ 를 구하는 것은 1차 대회 "오름차순" 문제와 유사하게 monotone stack을 관리하면 된다.

## 부분문제 6

센트로이드 분할을 이용해 문제를 해결한다. 센트로이드  $c$ 에 대해,  $c$ 를 지나는 모든 경로들을 처리한 후,  $c$ 를 제거하고 나뉜 각각의 트리에서 다시 문제를 해결하는 분할 정복 방식을 사용한다.

$c$ 를 기준으로  $x_1, \dots, x_k$ 의 서브트리로 나뉜다고 하자.  $x_1$ 의 서브트리의 속하는  $u$ 에 대한 답을 구해보자. 각  $v$ 에 대해  $u \rightarrow c$  경로와,  $c \rightarrow v$  경로로 분해해서 생각하자.

$u \rightarrow c$  경로에서  $c$ 에 도달했을 때의 점수 및 점수를 0으로 바꾼 횟수는 항상 일정하다. 이는 부분문제 5와 유사하게 동적 프로그래밍을 통해 구할 수 있는데, 이 서브태스크의 경우 트리의 높이가 낮으므로 점수가 0 미만이 되는 최초 지점을 부모를 타고 올라가면서 구해주면 충분하다. 이제 결과  $C$ 값은 모든  $v$ 에 대해 더해지므로, 트리의 크기  $M$ , 서브트리의 크기  $sz$ 에 대해  $M - sz_{x_1}$ 만큼 곱해서 답에 더해준다.

$c \rightarrow v$ 의 상의 정점  $w$ 에 대해,  $w$ 에서 점수가 0 미만이 되는 경우를 생각해보자.  $c$ 에서 점수가 0인 경우를 기준으로, 점수가 0 이상인 정점들은  $c$ 에서의 점수가 더 높아도 점수가 0 미만이 되지 않는다. 즉,  $c$ 에서 점수가 0인 경우 점수가 0 미만이 되는 정점들이  $w$ 의 후보가 된다. 그리고 어떤  $v$ 에 대한 답은 마지막  $w$ 부터의 거리이다. 이러한  $w$ 들은  $c$ 에서의 점수가 높아질수록  $c$ 에서 가까운 부분부터 점점 점수가 0 이상이 되게 된다. 구체적으로,  $c$ 와 거리  $D$ 에 대해  $c$ 에서의 점수가  $-D$  이상인 경우 점수가 0 이상이 된다.

두 경우를 종합하면 각  $u$ 에 대해  $c$ 에서의 점수를 구한 후, 해당 점수에 대한  $S$ 값과  $C$ 값을 스위핑을 통해 구할 수 있음을 알 수 있다. 시간복잡도는  $O(N \log^2 N)$ 이다.

## 부분문제 8

$u \rightarrow c$  경로의 값을 빠르게 구해야 한다. 세그먼트 트리에서의 이분 탐색, 스패스 테이블, 스택을 관리하되 pop할 정점들을 이분탐색으로 구하고 복원 연산을 구현하는 방법 등으로  $O(\log N)$ 에 구할 수 있다. 정해는 마지막 방법을 이용하여 구현되었다. 시간복잡도는  $O(N \log^2 N)$ 이다.