DRVTRUCK Solution:

Bài toán có thể diễn tả là với mỗi đỉnh, tìm hành trình đi đến k đỉnh được đánh dấu từ trước sao cho độ dài của hành trình là ngắn nhất.

Dễ thấy độ dài hành trình ngắn nhất với xe xuất phát từ đỉnh u bằng 2 ×(tổng khoảng cách từ u đến k điểm đánh đấu) - (độ dài đường đi dài nhất từ u đến một đỉnh được đánh dấu.

+) Tính tổng khoảng cách từ 1 đỉnh đến các đỉnh được đánh dấu

Gọi $T1[u]$ = tổng khoảng cách từ u đến các đỉnh được đánh dấu trong cây con gốc u

Gọi $T2[u]$= tổng khoảng cách từ u đến các đỉnh được đánh dấu ngoài cây con gốc u.

Ta có các công thức ($w = prev[u]$):

$$T1[u] = \sum\{T1[v] + S[v] * L(u,v): v \in con(u)\}$$

$$T2[u] = T2[w] + \big(T1[w] - T1[u] - S[u] * L(w,u)\big) + (k - S[u] * L(w,u)$$

Trong đó $S[u]$=số lượng dỉnh đánh dấu trong cây con gốc u.

Tổng khoảng cách từ các điểm đánh dấu đến u được tính bởi $T1[u] + T2[u]$

+) Tính đường đi dài nhất từ u đến một đỉnh được đánh dấu

Đặt $f[u], f2[u]$ là độ dài đường đi dài nhất, đường đi dài nhì từ một đỉnh đánh dấu thuộc cây con gốc u đến u ta có

$f[u] = f2[u] = -\infty$ nếu $S[u] = 0$

$f[u] = \max\big(0, f[v] + L(u,v): v \in con(u)\big)$ trong trường hợp ngược lại, $f2[u]$ được tính kèm khi tính $f[u]$.

Đặt $g[u]$ độ dài đường đi dài nhất từ u đến đỉnh đánh dấu không thuộc cây con gốc u. Ta có $g[u] = -\infty$ nếu $S[u] = k$ ngược lại:

$g[u] = \max(0, g[w] + L(w,u), f[w](hoặc\ f2[w]) + L(w,u)$

Độ dài đường đi dài nhất từ u đến một đỉnh được đánh dấu bằng $\max(f[u], g[u])$

Vậy với mỗi đỉnh u đáp số là:

$$2 \times (T1[u] + T2[u]) - \max(g[u], f[u])$$

Dễ thấy tất cả các công thức trên có thể tính sau khi có được một sắp xếp topo DFS trên cây.