



Descrierea soluției – drumuri

S.I. dr. ing. Mugurel Ionut Andreica – Universitatea „Politehnica” Bucuresti

Soluția 1 – 40 puncte

Se mențin drumurile sub forma de vectori, împreună cu informații asociate fiecărui nod referitor la drumul din care face parte și la poziția sa în drum. Adăugarea unei muchii presupune concatenarea a două drumuri (eventual în sens inversat), iar ștergerea unei muchii presupune separarea unui drum în două drumuri. Cu ajutorul acestor informații menținute, se poate verifica în timp $O(1)$ dacă adăugarea sau ștergerea unei muchii sunt valide, respectiv se poate răspunde în timp $O(1)$ la fiecare interogare (de ex., dacă nodul i se afla pe poziția x în drum, iar drumul este reținut ca un vector, atunci nodurile aflate la distanța D de nodul i se afla pe pozițiile $x-D$, respectiv $x+D$, din cadrul vectorului). Concatenarea a două drumuri și separarea unui drum în două sunt operații ce se execută în timp $O(N)$.

Soluția 2 – 40 puncte

Se menține graful folosind o reprezentare „standard” a grafului (de ex., liste de vecini). Pentru fiecare operație putem parcurge graful (în timp $O(N)$), căci există $O(N)$ noduri și muchii pentru a decide dacă operația se poate efectua, respectiv pentru a răspunde la interogare.



Solutia 3 – 40 puncte

Se mentin pentru fiecare nod cei maxim 2 vecini din graf. In plus, daca un nod este capat de drum, putem memora si capatul opus al aceluiasi drum. La adaugarea unei muchii setam cele doua noduri ca fiind vecine unul cu altul. Muchia este adaugata daca ambele noduri sunt capete de drumuri (au gradul cel mult egal cu 1) si nu sunt capetele aceluiasi drum. La stergerea unei muchii „resetam” vecinul corespunzator fiecaruia din cele doua noduri la „inexistent” (si, eventual, calculam capatul opus al fiecaruia din cele 2 noduri, care au devenit acum capete de drumuri).

Pentru a raspunde la o interogare vom pleca din nodul i si vom parcurge drumul din care face parte in ambele sensuri, ori pana ajungem la capatul sau, ori pana am parcurs o distanta D . Parcurgerea drumului presupune atentie la implementare, deoarece, in cadrul unui drum, nu putem mentine in mod consistent „vecinul stanga”, respectiv „vecinul dreapta” al fiecarui nod (decat daca efectuam procesari suplimentare la adaugarea unei muchii, adica la concatenarea a doua drumuri). Din fericire, mentinerea consistenta a vecinilor „stanga” si „dreapta” in cadrul unui drum nu este necesara. Cand parcurgem drumul intr-un sens, mentinem atat nodul curent, cat si nodul anterior. Apoi, urmatorul nod ce va fi parcurs este acel vecin al nodului curent care este diferit de cel anterior.

Aceasta solutie are complexitatea $O(N)$ pentru fiecare interogare si $O(1)$ pentru un tip de modificare a grafului si $O(N)$ pentru celalalt tip de modificare a grafului.



Solutia 4 – 100 puncte

Vom extinde solutia anterioara pentru a mentine, pentru fiecare nod, pe langa cei maxim 2 vecini, inca maxim 2 **K-vecini**, adica cele maxim 2 noduri aflate la distanta exact K de nodul respectiv. Daca notam cei doi vecini ai nodului i prin $v1[i]$ si $v2[i]$ si cei doi K-vecini ai nodului prin $kv1[i]$ si $kv2[i]$, vom avea proprietatea ca $kv1[i]$ este K-vecinul nodului i din acelasi sens ca si vecinul $v1[i]$ (si similar pentru $kv2[i]$ si $v2[i]$).

La adaugarea unei muchii maxim K noduri de fiecare parte a muchiei vor obtine K-vecini noi in sensul dinspre muchia adaugata. La stergerea unei muchii, maxim K noduri de fiecare parte a muchiei isi vor pierde K-vecinii din sensul inspre muchia stearsa.

Folosind K-vecinii, putem raspunde la interogari in timp $O(K+N/K)$. Mai intai vom „sari” din K in K, pana cand un nou salt de lungime K ar „sari” prea departe (ori peste distanta D, ori in afara drumului). Apoi vom continua sa ne deplasam din 1 in 1 (din vecin in vecin) pana ajungem la distanta D sau la capatul drumului. Si acest mod de parcurgere necesita aceeasi atentie la implementare din solutia anterioara (intrucat nu mentinem in mod consistent vecini si K-vecini „stanga” si „dreapta”, urmatorul vecin/K-vecin ce va fi parcurs este acel vecin/K-vecin al nodului curent care este diferit de vecinul/K-vecinul parcurs anterior).

Daca alegem $K=O(\sqrt{N})$, atunci complexitatea de timp a fiecarei operatii este $O(\sqrt{N})$.