

### ***Problema Tara Minunilor, student Airinei Adrian***

Asociem problemei un graf orientat cu  $N$  noduri, initial fiind muchie între oricare doua noduri consecutive. Trebuie sa facem urmatoarele observatii atunci cand introducem o muchie noua  $x \rightarrow y$  ( $x < y$ )

- daca nu exista o muchie  $a \rightarrow b$  introdusa anterior astfel incat  $a < x$  si  $b > y$  atunci raspunsul este  $x \cdot (N - y + 1)$
- daca exista o muchie  $a \rightarrow b$  introdusa anterior astfel incat  $a < x$  si  $b > y$ , fie  $a \rightarrow b$  muchia cu valoarea "a" maxima  
Atunci observam ca drumurile care isi schimba distanta minima între ele sunt perechi de forma  $(p, q)$  astfel
  1.  $p \leq x$  si  $y \leq q < b$
  2.  $a < p \leq x$  si  $q \geq y$
  3. trebuie sa scadem acum perechile unde  $a < p \leq x$  si  $y \leq q < b$  pentru ca le-am numarat de doua ori

Astfel raspunsul este  $x \cdot (b - y) + (x - a) \cdot (N - y + 1) - (x - a) \cdot (b - y)$ .

Deci trebuie sa aflam pentru fiecare muchie  $(x, y)$  introdusa muchia  $(a, b)$  despre care am facut referire anterior. Cea mai usoara modalitate de a face acest lucru este sa observam ca daca fiecarei muchii  $(x, y)$  i-am asocia o pereche de paranteze, o paranteza deschisa pentru  $x$  si una inchisa pentru  $y$  toate operatiile noastre ar determina la sfarsit un sir corect parantezat (asta datorita restrictiei impuse in enunt). Pentru a  $k$ -a operatie  $(x, y)$  vom introduce într-un vector doua triplete, unul  $(x, 0, k)$  si unul  $(y, 1, k)$  având semnificatia ca am introdus o paranteza deschisa la pozitia  $x$  si timpul  $k$  (timpul este indicele operatiei) si o paranteza inchisa la timpul  $y$  si timpul  $k$ . Vom sorta acest vector după prima valoare (corespunzătoare pozitiei). Atunci de fiecare data când apare un triplet  $(x, 0, k)$ , semn ca deschidem o paranteza, vrem sa stim dintre toate parantezele care sunt deschise dar nu au fost încă închise care este cea cu timpul mai mic decât  $k$ , cat mai tarziu introdusa in stiva. Acest lucru se poate face ușor folosind un arbore de intervale, doar cu operatiile de aflare a maximului pe un interval si modificare a unei valori. In arborele de intervale vom introduce pentru fiecare nod asociat unui timp  $x$  momentul la care a fost introdus in stiva. Astfel, trebuie sa aflam la un moment dat care nod din intervalul  $[1, a]$  are valoarea maxima. Dacă apare un triplet  $(y, 1, k)$  trebuie "sters" nodul respectiv din arborele de intervale. La sfarsit vom parcurgere din nou operatiile si vom afisa raspunsul pentru fiecare. Complexitatea va fi  $O(M \cdot \log M)$  timp si  $O(M)$  memorie. Mai exista alte solutii mai mult sau mai puțin complicate folosind diferite structuri de date, in complexitati  $O(M \cdot \log M)$  sau  $O(M \cdot \log M \cdot \log M)$ .