Problema Amazon

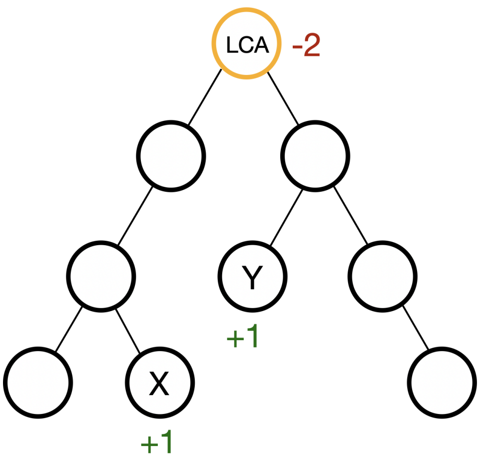
Clasele XI-XII
Autori Robert-Raul Borcani, Octavian Trifan, Daniel Todașcă
Universitatea Babeș Bolyai, Cluj-Napoca

Soluție de 20-35 puncte - O(M*N)

Se poate observa că graful dat este un arbore. Se stabilește nivelul fiecărui nod printr-un DFS, unde rădăcina are nivelul 1. Se construiește un vector de tați.
Pentru a rezolva problema, va trebui să scădem costul muchiilor care apar cel mai des în trasee (Greedy). Astfel, pentru fiecare muchie din graf, ținem un contor al numărului de apariții ale acelei muchii în trasee. Pentru fiecare pereche (X,Y) pentru care se dorește traseul, se parcurge drumul de la X la Y. La fiecare muchie întalnită, se crește contorul ei cu 1. De asemenea, se crește o sumă cu costul muchiilor întâlnite. Fiecare muchie poate fi echivalentă cu un nod, deoarece există o muchie unică de la nod la părintele lui. La final, se identifică muchiile cele mai frecvent întâlnite și se scade din sumă costul aferent operației.

Soluție de 100 puncte - O(NlogN + M)

Ineficiența soluției anterioare vine din faptul că pentru fiecare pereche trebuie să parcurgem traseul din arbore a perechii respective. Pentru a evita acest lucru, putem folosi o soluție asemănătoare cu șmenul lui Mars, dar adaptat pentru arborele nostru: un traseu între două noduri X și Y este format din traseul de la X până la LCA(X,Y) și de la LCA(X,Y) la Y. Astfel, pentru fiecare traseu, putem sa creștem un contor cu 1 in X și Y, și să îl scădem cu 2 în LCA(X,Y).



Numărul de apariții ale unui nod în trasee este suma tuturor contoarelor nodurilor din subarborele care îl are ca părinte pe nodul respectiv. În funcție de algoritmul ales pentru LCA, putem obține complexități de O(NlogN + MlogN) (LCA cu programare dinamică) sau O(NlogN + M) (LCA cu RMQ).