Solutie – 100p

O observatie importanta este ca toate sirurile date in input se vor retine ca o structura vector de <char, int>, char reprezentand caracterul si int numarul de repetitii. (de exemplu, abcccdd se va retine ca { (a,1), (b,1), (c,3), (d,2) }.

Folosind aceasta observatie verificam pentru fiecare string din query daca este subsir in sirurile vrednice si le retinem. (consideram acest vector W)

Acum avem 2 variante pentru rezolvarea de 100 de puncte. In cazul in care query ul este subsir, putem verifica brute force (depinzand de optimizarile la nivel de interpretare) de cate ori apare restrans in sirul vrednic restrans.

Altfel, vom tine cele N siruri date si toate sufixele acestora intr-un trie. (pentru abbccc vom tine abbccc, bbccc, ccc). Observam ca nu este nevoie sa tinem decat caracterele distincte (pentru abbccc tinem abc, bc si c) si in fiecare nod din trie un map in care v[i] = numarul de ori de care apare nodul respectiv in sirul i (din sirurile vrednice). Astfel, pentru string ul s, vom cauta in trie, iar cand ajungem in nodul corespunzator ultimului caracter din s, ans = suma ( map[ x ] ) , x = elementele din W (ne asiguram in modul acesta ca se respecta ambele reguli).

Dinu Cristian

University of Oxford