



Triaaj – descrierea soluției

student Cosmin-Mihai Tutunaru, Universitatea Babeș Bolyai

O primă idee este să normalizăm datele, deci astfel vom avea valori de la 1 la (cel mult) N .

În al doilea rând, putem transforma (virtual) fiecare din cele 1000 linii (fără prima) în două stive, câte una pentru fiecare parte. Deci vom avea 2000 de stive + prima linie.

Se observă că trebuie să sortăm N numere folosind cel mult $2 \cdot \sqrt{N}$ stive. Un simplu countsort nu merge deoarece numărul de numere (sau valoarea maximă) este mult mai mare decât memoria disponibilă.

Vom rezolva problema folosind o „generalizare” a algoritmului countsort:

Pentru fiecare vagon cu numărul X de pe prima linie, îl adăugăm pe stiva $(X \% 1000)$ din partea dreaptă.

Trecem pe la fiecare stivă X (din partea dreaptă, începând de la prima). Luăm fiecare element Y din ea (toate printr-o singură mutare) și adăugăm la stiva $Y/1000$ din partea stângă valoarea X .

Trecem pe la fiecare stivă X (din partea stângă, începând de la ultima). Luăm fiecare element Y din ea (tot pe toate dintr-o singură mutare) și îl adăugăm la prima linie prin partea stângă. Practic prin această ultimă parcurgere am luat toate vagoanele în ordine descrescătoare.

Complexitate: $O(N \log N)$ ca timp și $O(N)$ ca număr de vagoane mutate.