

## Descriere soluție – fsecv

Propunător : prof. Eugen Nodea  
Colegiul Național “Tudor Vladimirescu”, Târgu Jiu

### Câteva observații preliminare:

- interogările nu modifică șirul A
- dacă cunoaștem răspunsul la o interogare din intervalul  $[st, dr]$ , atunci putem deduce cu ușurință răspunsul la interogări din intervalele:  $[st+1, dr]$ ,  $[st-1, dr]$ ,  $[st, dr-1]$ .

Rezolvarea problemei are la baza **principiul** *square root decomposition*.

Vom preprocesa toate interogările astfel încât rezultatul unei interogări să poată fi utilizat în următoarea interogare.

### Descriere algoritm:

1. partiționăm / împărțim în blocuri/bucăți de lungime  $k = \sqrt{N}$  șirul A
2. sortăm interogările după capetele intervalelor. Altfel spus, dacă avem interogările  $[L1, R1]$  și  $[L2, R2]$ , criteriile de sortare sunt:
  - $L1 / k < L2 / k$
  - $L1 / k == L2 / k \ \&\& \ R1 < R2$
3. parcurgem interogările sortate astfel încât, pe cât posibil interogarea curentă să permită folosirea informații din interogarea precedentă:

De exemplu, dacă interogarea anterioară este  $[1, 7]$  iar interogarea curentă este  $[4, 9]$ , atunci vom actualiza prin scădere (**scoate**) frecvența de apariții pt. elemente distincte din secvența  $A[1], \dots, A[4]$  și vom actualiza prin adăugare (**add**) frecvența de apariții pt. elemente distincte din secvența  $A[5], \dots, A[9]$ .

Altfel spus, dacă dorim să răspundem la interogarea  $[L, R]$  iar în prealabil am răspuns / procesat interogarea  $[st, dr]$ , atunci:

```
while (st > L) add(--st);  
while (dr < R) add(++dr);  
while (st < L) scoate(st++);  
while (dr > R) scoate(dr--);
```

Complexitate sortare:  $O(q * \log q)$

Complexitate procesare interogări:  $O((q+n) * \sqrt{n})$