

## Sort

### Descrierea soluției

#### Soluția 1.

Se pot construi doi vectori, unul care reține datele din fișier, un al doilea care să fie ordonat, fie aceștia **a[]** și respectiv **b[]**.

Se studiază în prima parte cazurile favorabile, acelea care necesită doar o schimbare pentru a poziționa valoarea **a[i]** în categoria finală, prin categorii înțelegând grupa valorilor 1, 2 sau 3.

Astfel dacă elementul **a[i1]** este în categoria  $c_k$  și ar trebui să fie în categoria  $c_j$  iar elementul **a[i2]** este în categoria  $c_j$  și ar trebui să fie în categoria  $c_k$  acesta este un caz favorabil necesită o schimbare.

Se realizează în cadrul acestui studiu și modificarea vectorului **a[]** prin schimbările permise.

Astfel pentru

**a[]** = {1, 3, 1, 2, 2, 3, 1, 3, 2, 1} și

**b[]** = {1, 1, 1, 1, 2, 2, 2, 3, 3, 3}

situație favorabilă ar fi de exemplu, pentru **a[2] = 3** și **a[10] = 1**.

Elementele greșit plasate, rămase, formează cicluri de elemente: 1 trebuie să fie 2, 2 trebuie să fie 3, iar 3 trebuie să fie 1. Avem nevoie de două schimbări pentru a corecta un astfel de ciclu. Astfel vom calcula numărul unor astfel de cicluri (calculând elementele greșit plasate și împărțind cu trei) iar rezultatul înmulțindu-l cu doi.

Adăugând acest număr la valoarea obținută anterior rezultă numărul total de schimbări.

#### Soluția 2.

Se determină câte valori 1, 2, respectiv 3 sunt în vector.

În zona valorilor 1, se numără câte valori 2 (**v12**) și câte valori 3 (**v13**) sunt pe pozițiile valorilor 1. Analog pentru zona ce ar trebui ocupată cu valori 2, se determină câte valori 1 (**v21**) și câte valori 3 (**v23**) sunt aici și la fel pentru **v31** și **v32**.

Numărul cazurilor favorabile va fi dat de

**min(v21, v12) + min(v31, v13) + min(v23, v32).**

Cazurile defavorabile vor fi:

**2 \* (max(v12, v21) - min(v21, v12));**

Prof. Maria și Adrian NIȚĂ