Algoritmi şi structuri de date (I). Seminar 10: Aplicaţii ale tehnicii divizării. Aplicaţii ale interclasării.

Problema 1 Fie $a[1..m]$ şi $b[1..n]$ două tablouri ordonate crescător având elemente nu neapărat distincte. Propuneţi un algoritm de complexitate $\mathcal{O}(max(m,n))$ care construieşte un tablou strict crescător ce conţine elementele distincte din a şi b.

Indicaţie. Se aplică tehnica interclasării verificând de fiecare dată la completarea în tabloul destinaţie dacă elementul ce ar trebui adăugat este diferit de ultimul element din tablou.

Problema 2 Se consideră două numere întregi date prin tablourile corespunzătoare descompunerilor lor în factori primi. Să se construiască tablourile similare corespunzătoare celui mai mic multiplu comun al celor două valori.

Indicaţie. Să considerăm numerele: $3415 = 3 \cdot 5^3 \cdot 7 \cdot 13$ şi $966280 = 2^3 \cdot 5 \cdot 7^2 \cdot 17 \cdot 29$. Primului număr îi corespund tablourile: $(3, 5, 7, 13)$ respectiv $(1, 3, 1, 1)$ iar celui de al doilea număr îi corespund tablourile $(2, 5, 7, 17, 29)$ respectiv $(3, 1, 2, 1, 1)$.

Tablourile corespunzătoare celui mai mic multiplu comun vor fi:

- *factori primi:* tabloul care conţine toţi factorii primi corespunzători celor două numere şi care poate fi obţinut prin interclasare ţinând cont că aceştia trebuie să fie distincţi.
- *exponenţi:* tabloul ce conţine valorile maxime ale exponenţilor.

În cazul exemplului cele două tablouri sunt: $(2, 3, 5, 7, 13, 17, 29)$ respectiv $(3, 1, 3, 2, 1, 1, 1)$

Problema 3 *Problema selecţiei celui de al k-lea element.* Fie $a[1..n]$ un tablou, nu neapărat ordonat. Să se determine al k-lea element al tabloului, selectat în ordine crescătoare (pentru $k = 1$ se obţine minimul, pentru $k = n$ se obţine maximul etc.).

Indicaţie. O primă variantă de rezolvare o reprezintă ordonarea crescătoare a tabloului (prin metoda selecţiei) până ajung ordonate primele k elemente ale tabloului. Numărul de operaţii efectuate în acest caz este proporţional cu kn. Pentru k apropiat de $n/2$ aceasta conduce la un algoritm de complexitate pătratică. Un algoritm de complexitate mai mică se obţine aplicând strategia de partiţionare de la sortarea rapidă: folosind o valoare de referinţă (de exemplu cea aflată pe prima poziţie în tablou) se partiţionează tabloul în două subtablouri astfel încât elementele aflate în primul subtablou să fie toate mai mici decât valoarea de referinţă iar elementele din al doilea subtablou să fie toate mai mari decât valoarea de referinţă.

Problema 4 *Determinare mediană.* Mediana unui tablou cu n elemente este elementul aflat pe poziţia $\lfloor (n+1)/2 \rfloor$ în cadrul tabloului ordonat crescător (obs: în cazul în care n este par se consideră uneori ca mediană media aritmetică a valorilor aflate în mijlocul tabloului). Fie $x[1..n]$ şi $y[1..n]$ două tablouri ordonate crescător. Să se determine mediana tabloului $z[1..2n]$ care conţine toate elementele din x şi y.

Indicaţie. O primă variantă ar fi să se construiască tabloul z prin interclasare şi să se returneze elementul de pe poziţia n. Este însă suficient să se realizeze o interclasare parţială până când se obţine elementul de poziţia n. În acest scop se poate folosi interclasarea bazată pe valori santinelă mai mari decât elementele din ambele tablouri.

Probleme suplimentare

1. Fie $a[1..m]$ şi $b[1..n]$ două tablouri ordonate strict crescător. Propuneţi un algoritm de complexitate liniară pentru determinarea mulţimii elementelor comune celor două tablouri.
   Indicaţie. Se utilizează ideea de la interclasare însă se reţin doar valorile identice din cele două tablouri.

2. Se consideră două numere întregi date prin tablourile corespunzătoare descompunerilor lor în factori primi. Să se construiască tablourile similare corespunzătoare celui mai mare divizor comun al celor două valori.

3. Se consideră trei tablouri $a[1..m]$, $b[1..n]$ şi $c[1..p]$ cu proprietăîle: a şi b sunt ordonate strict crescător iar c este ordonat strict descrescător. Propuneţi un algoritm de complexitate liniară în raport cu dimensiunile celor trei tablouri care construieşte tabloul crescător $d[1..m + n + p]$ care conţine toate elementele tablourilor a, b şi c.

4. Propuneţi un algoritm de complexitate $\mathcal{O}(n \log n)$ pentru a determina numărul de perechi $(a[i], a[j])$ ale unui tablou $a[1..n]$ (tabloul are elemente distincte) având proprietatea că $i < j$ şi $a[i] > a[j]$.
Indicaţie. Se foloseşte ideea de la sortarea prin interclasare doar că nu se modifică tabloul ci doar se numără de câte ori o valoarea mai mare se află înaintea unei valori mai mici.

5. Se consideră un tablou $a[1..n]$ care conţine numere întregi (atât pozitive cât şi negative). Propuneţi un algoritm de complexitate liniară (şi care foloseşte spaţiu suplimentar de dimensiune constantă) pentru a transforma tabloul iniţial astfel încât toate valorile negative să fie înaintea celor pozitive.
Indicaţie. Se poate utiliza ideaa de la determinarea unei poziţii de partiţionare folosind 0 ca valoare de referinţă.

6. Se consideră un şir de caractere $a[1..n]$ care conţine simboluri din mulţimea $\{a, b, c\}$. Propuneţi un algoritm de complexitate liniară care transformă şirul a astfel încât toate simbolurile "a" să fie grupate la începutul şirului, toate simbolurile "c" să fie la sfârşitul şirului iar simbolurile "b" să fie în mijlocul şirului.

7. Implementaţi metoda Strassen pentru calculul produsului a două matrici pătratice $A[1..n, 1..n]$ şi $B[1..n, 1..n]$ în ipoteza că n este o putere a lui 2 (pentru detalii privind metoda Strassen se poate consulta https://walkccc.me/CLRS/Chap04/4.2/).