

## Rezolvarea eficientă a unor probleme de algoritmică



### Obiective

Rezolvarea problemelor folosind metode diferite din punct de vedere al complexității (temporale și spațiale).



### Aspecte teoretice

Rezolvarea problemelor.  
Analiza complexității procesului de rezolvare.



### Termen de predare și evaluare

Laborator 2

#### Obligatoriu:

5 probleme la alegere, pentru fiecare problemă se va prezenta o soluție. Fiecare rezolvare corectă va primi 20 puncte.

#### Opțional:

Rezolvarea prin alte metode a problemelor alese la punctul precedent sau rezolvarea unor noi probleme. Fiecare rezolvare corectă va primi 20 puncte.

**Notă:** punctajul maxim acumulat pentru acest laborator este 200 puncte.

### Cerințe



**Live:** minim 2 probleme (din cele 5)

**Temă:** restul de probleme

Specificați, implementați și testați subalgoritmi pentru următoarele cerințe. Încercați să rezolvați fiecare cerință cât mai eficient (ca număr de pași și / sau ca resurse de memorie folosite).

1. Să se determine ultimul (din punct de vedere alfabetic) cuvânt care poate apărea într-un text care conține mai multe cuvinte separate prin " " (spațiu).  
De ex. ultimul (dpdv alfabetic) cuvânt din "Ana are mere rosii si galbene" este cuvântul "si".
2. Să se determine distanța Euclideană între două locații identificate prin perechi de numere.  
De ex. distanța între (1,5) și (4,1) este 5.0
3. Să se determine produsul scalar a doi vectori rari care conțin numere reale. Un vector este rar atunci când conține multe elemente nule. Vectorii pot avea oricâte dimensiuni.  
De ex. produsul scalar a 2 vectori unisimensionali [1,0,2,0,3] și [1,2,0,3,1] este 4.
4. Să se determine cuvintele unui text care apar exact o singură dată în acel text.  
De ex. cuvintele care apar o singură dată în "ana are ana are mere rosii ana" sunt: 'mere' și 'rosii'

5. Pentru un șir cu  $n$  elemente care conține valori din mulțimea  $\{1, 2, \dots, n - 1\}$  astfel încât o singură valoare se repetă de două ori, să se identifice acea valoare care se repetă.

De ex. în șirul  $[1, 2, 3, 4, 2]$  valoarea 2 apare de două ori.

6. Pentru un șir cu  $n$  numere întregi care conține și duplicate, să se determine elementul majoritar (care apare de mai mult de  $n / 2$  ori).

De ex. 2 este elementul majoritar în șirul  $[2, 8, 7, 2, 2, 5, 2, 3, 1, 2, 2]$ .

7. Să se determine al  $k$ -lea cel mai mare element al unui șir de numere cu  $n$  elemente ( $k < n$ ).

De ex. al 2-lea cel mai mare element din șirul  $[7, 4, 6, 3, 9, 1]$  este 7.

8. Să se genereze toate numerele (în reprezentare binară) cuprinse între 1 și  $n$ .

De ex. dacă  $n = 4$ , numerele sunt: 1, 10, 11, 100.

9. Considerându-se o matrice cu  $n \times m$  elemente întregi și o listă cu perechi formate din coordonatele a 2 căsuțe din matrice ( $(p, q)$  și  $(r, s)$ ), să se calculeze suma elementelor din sub-matricile identificate de fiecare pereche.

De ex, pt matricea

```
[ [0, 2, 5, 4, 1],
  [4, 8, 2, 3, 7],
  [6, 3, 4, 6, 2],
  [7, 3, 1, 8, 3],
  [1, 5, 7, 9, 4]]
```

și lista de perechi  $((1, 1)$  și  $(3, 3))$ ,  $((2, 2)$  și  $(4, 4))$ , suma elementelor din prima sub-matrice este 38, iar suma elementelor din a 2-a sub-matrice este 44.

10. Considerându-se o matrice cu  $n \times m$  elemente binare (0 sau 1) sortate crescător pe linii, să se identifice indexul liniei care conține cele mai multe elemente de 1.

De ex. în matricea

```
[ [0, 0, 0, 1, 1],
  [0, 1, 1, 1, 1],
  [0, 0, 1, 1, 1]]
```

a doua linie conține cele mai multe elemente 1.

11. Considerându-se o matrice cu  $n \times m$  elemente binare (0 sau 1), să se înlocuiască cu 1 toate aparițiile elementelor egale cu 0 care sunt complet înconjurate de 1.

De ex. matricea

devine

```
[ [1, 1, 1, 1, 0, 0, 1, 1, 0, 1],
  [1, 0, 0, 1, 1, 0, 1, 1, 1, 1],
  [1, 0, 0, 1, 1, 1, 1, 1, 1, 1],
  [1, 1, 1, 1, 0, 0, 1, 1, 0, 1],
  [1, 0, 0, 1, 1, 0, 1, 1, 0, 0],
  [1, 1, 0, 1, 1, 0, 0, 1, 0, 1],
  [1, 1, 1, 0, 1, 0, 1, 0, 0, 1],
  [1, 1, 1, 0, 1, 1, 1, 1, 1, 1]]
```

```
[ [1, 1, 1, 1, 0, 0, 1, 1, 0, 1],
  [1, 1, 1, 1, 1, 0, 1, 1, 1, 1],
  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1],
  [1, 1, 1, 1, 1, 1, 1, 1, 0, 1],
  [1, 1, 1, 1, 1, 1, 1, 1, 0, 0],
  [1, 1, 1, 1, 1, 1, 1, 1, 0, 1],
  [1, 1, 1, 0, 1, 1, 1, 0, 0, 1],
  [1, 1, 1, 0, 1, 1, 1, 1, 1, 1]]
```