CURS 8

TEHNICI DE PROGRAMARE

METODA GREEDY

- Exemplu: $A = \{5, 1, -7, 3, -2, -1, 10\} \Rightarrow S = \{5, 1, 3, 10\}$

  $S = ?$ a.î. $\sum_{x \in S} x$ să fie max ; $S \subseteq A$

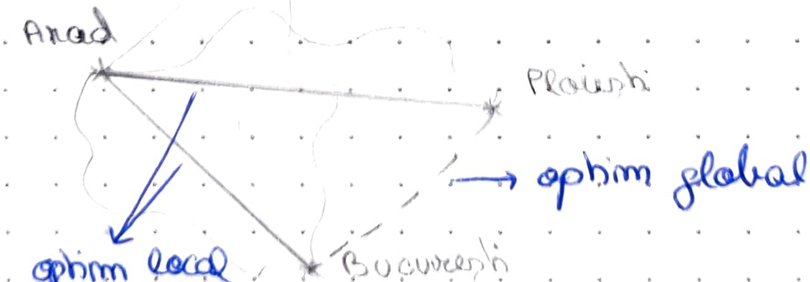
- Exemplu: $A = \{-5, -1, -7, -3\} \Rightarrow S = \{-1\}$

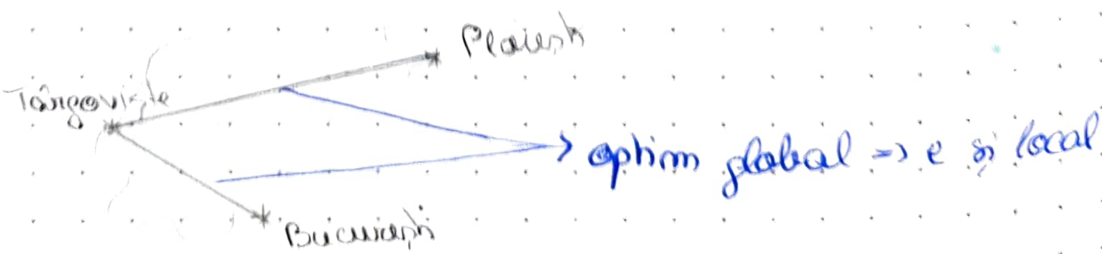
- Forma generală a unei probleme de optimizare:

  Fie $A \neq \emptyset$. Să se det o submulț. $S \subseteq A$ a.î. valoarea unei funcții obiectiv $f: \mathcal{P}(A) \to \mathbb{R}$ să fie min/max pe S

  max pe $S \Leftrightarrow \forall S' \subseteq A,\ S' \neq S$ avem $f(S') \leq f(S)$

- Principiul optim:
  - Optimul global implică local (da)
  - Optime locale $\Longrightarrow$ optim global (depinde)





Forma generală a unui algoritm de tip Greedy

- Se stabilește un criteriu de selecție pt. ca un elem din A să fie adă. în sol. S și se dem. matematic corectitudinea sa
- Se sortează convenabil elem din A (opțional)

Exemplu: $A = \{5, -1, 7, -3, -8, 9, 2, -4\}$; $S = ?$ a.î. $\sum_{x \in S} x$ max

$k = |S| = 5$

Var 1: $O(km) \rightarrow O(m)$ sau $O(m^2)$; $1 \le k \le m$

Var 2: $O(m \cdot \log_2 m + k) \simeq O(m \log_2 m)$

sortare ; $1 \le k \le m$; $A = \{9, 7, 5, 2, -1, -3, -7, -8\}$

- Se parcurge A elem. cu elem și se selectează în S elem. care verif. criteriul

```
S ← ∅
for x in A:
    if criteriu(x) == True:
        S ← S ∪ {x}
```

Complexitate:

$O(m)$, $O(m \log_2 m)$, $O(m^2)$
↳ sort   ↳ sort
$O(m \cdot O(\text{crit}))$

Contraexemplu: Plata unei sume folosind un nr. min. monede

- $v = \{8, 7, 1\}$ €

  $S = 14€ = 2 \cdot 8€ + 1€ = 2 \times 8€ + 1 \times 1€ \Rightarrow 3$ mon.

- $S = 14€ = 1 \times 8€ + 6 \times 1€ \Rightarrow 7$ mon. (nu e optim)
- $v = \{8, 7, 5\}$

  $S = 14€ = 1 \times 8 + 1 \times 5 + \underbrace{1€}_{\text{rest neplatibil}} \Rightarrow$ pb. nu are rezolvare (∄)
  ↳ $2 \times 7€$

Minimizarea timpului mediu de așteptare

1 ghișeu $\longleftarrow$ 1 coadă cu m persoane

Pt cele m pers cunoaștem timpii individuali de servire

$t_1, t_2, t_3, \dots, t_m$

Să se găsească o modalitate de rearanjare a pers. la coadă, a.î. timpul de așteptare să fie min.

$m = 7$ (timpii individuali de serv)

$t = (5, 1, 7, 3, 5, 2, 3)$

$t_i = (5, 6, 13, 16, 21, 23, 26) \Rightarrow t_m = \frac{110}{7} = 15{,}7$ minute

$t' = (1, 2, 3, 3, 5, 5, 7)$

$t'_i = (1, 3, 6, 9, 14, 19, 26) \Rightarrow t'_m = \frac{78}{7} = 11{,}1$ minute

t_1 ar trb să aștepte 5 min	o să aștepte 5 min.
t_2 ar trb să aștepte 1 mi	o să aștepte $t_1 = 5 + 1$ min $= 6$
t_3 ar trb să aștepte 7 min	o să aștepte $t_1 + t_2 + 7 = 13$ etc.

Deci se ord crescător pt. a aștepta fiecare (cât mai puțin)

Dem: Pp că sunt așezate $p_1, p_2, \dots, p_m$ a.î. $t_1 \le t_2 \le \dots \le t_i \le t_j \le \dots \le t_m$

Pp prin abs. că sol nu e optim $\Rightarrow \exists$ o sol. optimă

$$t_{m_1} = \frac{t_1 + (t_1 + t_2) + \dots + (t_1 + \dots + t_i) + \dots + (t_1 + \dots + t_m)}{m}$$

$$t_{m_2} = \frac{m t_1 + (m-1) t_2 + \dots + (m-i+1) t_i + \dots + t_m}{m}$$

$$t_{m_1} - t_{m_2} = \frac{(m-i+1) t_i + (m-j+1) t_j - (m-i+1) t_j - (m-j+1) t_i}{m}$$

$$t_{m_1} - t_{m_2} = \frac{t_i (j-i) + t_j (i-j)}{m} = \frac{(i-j)(t_i - t_j)}{m} < 0$$