Orice algoritm care calculează $\begin{cases} m \\ M \end{cases}$ necesită $\geq \left[\frac{3n}{2}\right]-2$ comp

0) Există algoritm care rezolvă problema

→ Există probleme pentru care nu se pot elabora algoritme

Problema terminării programului nu este decidabilă

(nu există alg. pentru ea)

Problema Echivalenței programului

Intrare: 2 programe

Ieșire: Sunt echivalente sau nu

Informatică

Dem Existențială

$\exists\, \alpha, \beta \in \mathbb{Q}$ cu $\alpha^{\beta} \in \mathbb{Q}$

Problemă P

0) $\exists$ algoritmul ce rezolvă P

1) Creeam algoritmul

2) $T(n)$ n = nr. date de intrare

3) Corectitudine

4) Optimalitate

$n = 2K$

$T(n) = \frac{3n}{2} - 2$ comp $= 3K-2$

Optim?

a_i a_j | Max → cel care bate pe toți

| min → cel care este bătut de toți

$\{a_1, \dots, a_n\}$ m: min a_i, $i=\overline{1,n}$ M: max a_i, $i=\overline{1,n}$

$n-1$ comp pentru min
$n-1$ comp pentru max } $2n-2$ comp

$m = a_1$ $M = a_1$

for $i = \overline{2,n}$
  if $a_i < m$ then $m = a_i$
  if $a_i > M$ then $M = a_i$
end for

$\underbrace{a_1, \dots, a_K}_{m, M}, \underbrace{a_{K+1}, a_{K+2}}_{\cancel{4}\ 3 comp}, \dots, a_n$

A = mulțimea jucătorilor care n-au jucat

B = " " " au jucat și câștigat

C = " " " " " " pierdut

D = " " " " " " câștigat și pierdut

Starea inițială $(n, 0, 0, 0)$

finală $(\underset{a}{0}, \underset{b}{1}, \underset{c}{1}, \underset{d}{n-2})$

A A $\rightarrow$ $(a-2, b+1, c+1, d)$

B B $\rightarrow$ $(a, b-1, c, d+1$

C C $\rightarrow$ $(a, b, c-1, d+1)$

A B $\rightarrow$ $\cancel{(a-1, b, c, d+1)}$ $a>b$

$(a-1, b, c+1, d)$ $a<b$

A C $\rightarrow$ $(a-1, b+1, c, d)$ $a>c$

$\cancel{(a-1, b, c, d+1)}$ $a<c$

B C $\rightarrow$ (a, b, c, d) $b>c$

$(a, b-1, c-1, d+2)$ $b<c$

Cel mai rapid rezultat

$(2k, 0, 0, 0) \xrightarrow[\text{A cu A}]{k \text{ ori}} (0, k, k, 0) \xrightarrow[\text{B cu B}]{k-1 \text{ ori}} (0, 1, k, k-1)$

$\xrightarrow[\text{B cu C}]{k-1 \text{ ori}} (0, 1, 1, 2k-1)$

Metoda Greedy

Mulțime cu $\boxed{n \text{ elemente}}$ are $\boxed{2^n \text{ soluții}}$

A finită

Optimizează $\begin{cases} \text{minim} \\ \text{maxim} \end{cases}$

$p: \mathcal{P}(A) \rightarrow \{0, 1\}$

$x \subset A$ sol dacă $p(x) = 1$

$f: \mathcal{P}(A) \rightarrow R_+$

Caut soluții (subcodat $p(x) = 1$)

care optimizează f

```
prelucrare(A) ; S ← ∅
for i = 1,n
    if p(S ∪ {ai}) = 1
        then S ← S ∪ {ai}
```

ordonare $O(n \cdot \log n)$

Vreau maxim
↳ fiecare pas aleg max. local

rezultatul este $\underline{\underline{S}}$ → nu a apărut funcția f

---

A finită

Caut submulțimea de un anumit tip care optimizează o funcție de cost

- Submulțime acceptabilă
- Submulțime ale cărei elemente se "înțeleg bine" între ele

$\{a_1, \dots, a_n\}$ $\mathbb{Z}$

Caută elemente de sumă

```
S = ∅;
for i : 1,n
    if ai > 0 then S ← S ∪ {ai}
```

$(a_1, \dots, a_n)$

Caut cel mai lung șir struct ↑ | Caut submulțimea de sumă maximă, dar ≤ M

Prel. sortez ↑ elemente lui a | Prel 6, 1, 3 M = 7

1, 1, 2, 3, 3, 3, 4, 4, 4, 4, 7

2

2

3

2

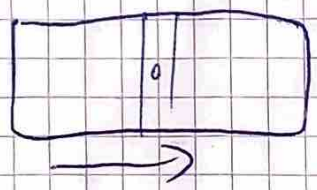
Timp foarte bun de execuție $O(n \cdot \log n)$

Rezultat: nu e neapărat corect

Trebuie o dem.

$f(S) \to \min$
$\quad \hookrightarrow \max$

Memorarea totală pe brandă



$L(1), \dots, L(n)$ lungime textilă

Sol = permutare p de ordin n

$P = (P_1, \dots, P_n)$

1	2	3	4
3	7	4	1

$p = (3, 1, 4, 2)$

4	3	1	7

$T_p(k) = \sum_{i=1}^{k} L(P_i) \quad \forall k = 1, \dots, n \qquad T_p(3) = 8$

citirea informației de pe poz. k

Probabilitatea = texte

$T(p) = \frac{1}{n} \sum_{k=1}^{n} T_p(k) \to$ Caut p care minimizează T

<u>Prelucrăm</u> ordonăm ~~1~~ textele după lungimea L

<u>Aleg sol</u> $p = id$

<u>Dem</u> că id este sol optimă

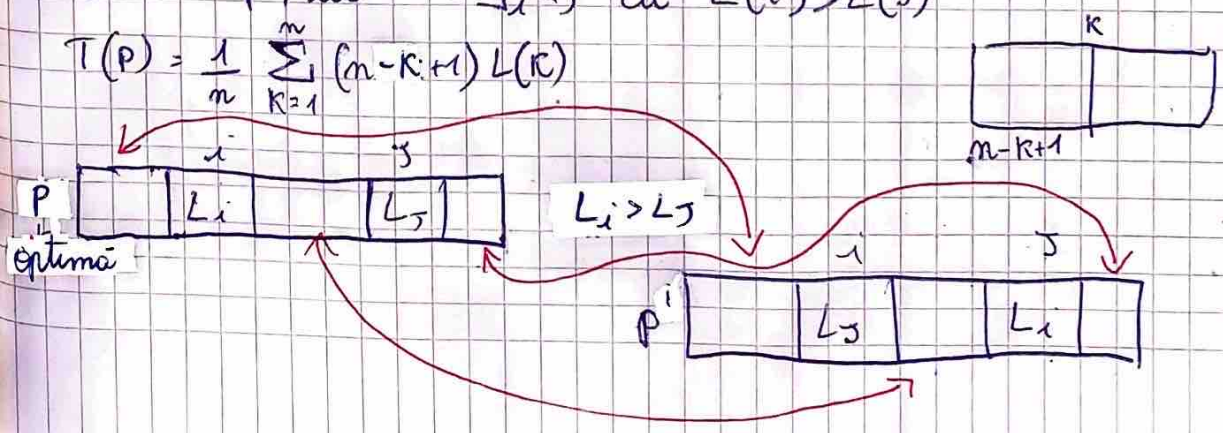
$T(id) \le T(p) \quad \forall p$ permutare de gradul n

Fie p - permutare optimă

Dacă $p = id$ (identică) $\exists i < j \quad L(i) < L(j)$

Dacă $p \ne id \to \exists i < j$ cu $L(i) > L(j)$

$T(p) = \frac{1}{n} \sum_{k=1}^{n} (n-k+1) L(k)$



ERROR 404

$$n\left[T(p) - T(p')\right] = (n-i+1)L(i) + (n-j+1)L(j) - (n-i+1)L(j) - (n-j+1)L(i)$$

$$= (j-i)L(i) + (i-j)L(j) = (j-i)\left[L(i) - L(j)\right]$$

$L_i > L_j \rightarrow T(p) - T(p') > 0 \rightarrow T(p) \geq T(p')$ CONTRADICTIE