Editační vzdálenost

koule → kdoule, kule, boule poutník → potemník

Df:

Editační operace: změna, vložení, smazání } 1 znaku

Edit. vzdálenost řetězců α, β := min. délka posloupnosti edit. operací, které z α udělají β } $L(\alpha,\beta)$

- též Levenštejnova 1965
- je to metrika

operace jsou BÚNO prováděny zleva doprava v α

jak vypadá první operace

$\alpha = a_1 \ldots a_n$ $\beta = b_1 \ldots b_m$

$L(\alpha,\beta)$

1. smaže a_1 $1 + L(a_2 \ldots a_n, \beta)$
2. změní a_1 na b_1 $1 + L(a_2 \ldots a_n, b_2 \ldots b_m)$
3. vloží b_1 před a_1 $1 + L(\alpha, b_2 \ldots b_m)$
4. ponecháme a_1 i b_1 $L(a_2 \ldots a_n, b_2 \ldots b_m)$ pokud $a_1 = b_1$

zkusíme 1–4, vybereme min.

Edit(i, j): ← spočítá $L(\alpha_i \ldots \alpha_n, \beta_j \ldots \beta_m)$

1. Pokud $i > n$: Vrátíme $m - j + 1$
   Pokud $j > m$: Vrátíme $n - i + 1$
2. $\ell_v \leftarrow 1 + \mathrm{Edit}(i, j+1)$ ③
3. $\ell_s \leftarrow 1 + \mathrm{Edit}(i+1, j)$ ①
4. $\ell_z \leftarrow \mathrm{Edit}(i+1, j+1)$ ② nebo ④
5. Pokud $\alpha_i \neq \beta_j$: $\ell_z \leftarrow \ell_z + 1$
6. Vrátíme $\min(\ell_v, \ell_s, \ell_z)$

Pomalé pro $a \ldots a$, $a \ldots a$

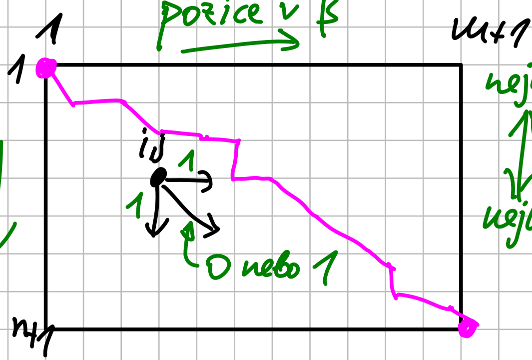
$i \in \{1, \ldots, n+1\}$, $j \in \{1, \ldots, m+1\}$ → jen $\Theta(n \cdot m)$ různých volání → kešování → čas $\Theta(mn)$

Nerekurzivní verze:

1. Pro $j = 1, \ldots, m+1$: $T[n+1, j] \leftarrow m - j + 1$
   Pro $i = 1, \ldots, n$: $T[i, m+1] \leftarrow n - i + 1$
2. Pro $i = n, \ldots, 1$:
3. Pro $j = m, \ldots, 1$:
4. $\delta \leftarrow \begin{cases} 0 & \text{pokud } \alpha_i = \beta_j \\ 1 \end{cases}$
5. $T[i,j] = \min(1 + T[i+1, j], 1 + T[i, j+1], \delta + T[i+1, j+1])$

Čas. složitost $\Theta(nm)$
prostor. složitost $\Theta(nm)$ → lze zlepšit na $\Theta(n+m)$

Grafový pohled:

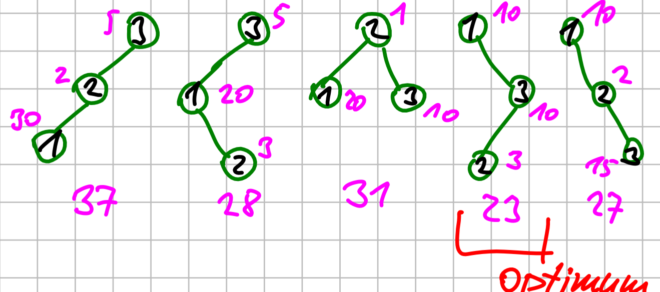


nejkratší cesta z (1,1) do (n,m) ↔ nejkratší posl. edit. operací, která z α udělá β (uspořádáno zleva doprava)

Vytvoříme graf $\Theta(nm)$ → nejkratší cesta v DAGu $\Theta(nm)$ → edit. vzdálenost } $\Theta(nm)$

Optimální BVS

na 1 se ptáme 10×
2 1×
3 5×



Problém: Dány klíče $x_1 < \ldots < x_n$
a váhy $w_1 \ldots w_n \in \mathbb{N}$

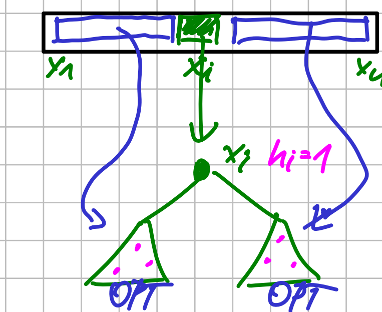
Pro BVS T na $x_1 \ldots x_n$: hloubky $h_1 \ldots h_n$

h_i = # vrcholů na cestě kořen → x_i

$C(T) := \sum_i h_i \cdot w_i$
cena

Cíl: Najít T s min. $C(T)$.

Co kdyby v kořeni opt. stromu bylo x_i



$OPT(x_1 \ldots x_n) = OPT(x_1 \ldots x_{i-1}) + OPT(x_{i+1} \ldots x_n) + \sum_{j=1}^{n} w_j$

↳ my ale kořen neznáme
⇒ zkusíme $x_1 \ldots x_n$
a najdeme min. ceny

$OptBVS(\ell, p)$: ← opt. cena BVS pro $x_\ell \ldots x_p$

1. Pokud $\ell > p$: vrátíme 0.
2. vrátíme $\min(C_\ell \ldots C_p) + \sum_{i=\ell}^{p} w_i$
   kde $C_i := OptBVS(\ell, i-1) + OptBVS(i+1, p)$

① je to pomalé
② $\ell, p \in \{1 \ldots n+1\}$ → $O(n^2)$ podproblémů
③ kešování → počítáme $O(n^2)$ pp., každý v čase $O(n)$ } celkem $O(n^3)$
④ nahradíme rekurzi cyklem … od nejkratších intervalů k nejdelším

1. Pro $\ell = 1, \ldots, n$: $T[\ell, \ell-1] \leftarrow 0$
2. Pro $d = 1, \ldots, n$: d = délka intervalu
3. Pro $\ell = 1, \ldots, n-d+1$: ℓ = levý okraj
4. $p \leftarrow \ell + d - 1$ p = pravý okraj
5. $T[\ell, p] \leftarrow \min(C_\ell \ldots C_p) + \sum_{i=\ell}^{p} w_i$
   $C_i := T[\ell, i-1] + T[i+1, p]$
6. Vrátíme $T[1, n]$

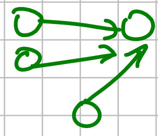
čas $O(n^3)$
prostor $O(n^2)$

⑤ cenu máme, ale jak vypadá strom? → tady si zapamatujeme, pro které C_i se nabylo minima
↳ to je kořen opt. stromu pro interval $[\ell, p]$

↓ rekurzivní budování (jako u dokonale vyváženého stromu) využívající zapamatované kořeny $O(n)$

Na okraj: umí se to i v čase $O(n^2)$. Trik: $kořen[\ell, p-1] \le kořen[\ell, p] \le kořen[\ell+1, p]$ ← Knuthova nerovnost

DP obecně: Systém podproblémů (stavy DP)
a závislostí mezi nimi
↑
tvoří DAG
Procházíme stavy v topologickém pořadí.



Příklad: Máme orientovaný graf s vrcholy $\{1 \ldots n\}$ bez záporných cyklů
a matici délek hran $L \in \mathbb{R}^{n \times n}$:
$$L_{ij} = \begin{cases} \text{délka hrany } (i,j) \\ +\infty \text{ pokud hrana neexistuje} \\ 0 \text{ pro } i=j \end{cases}$$

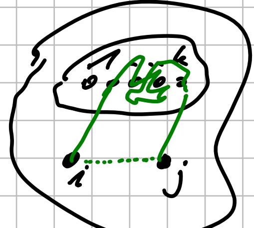
Chceme matici vzdáleností $D \in \mathbb{R}^{n \times n}$:
D_{ij} := délka nejkr. cesty z i do j
speciálně $D_{ii} = 0$

Už umíme: $n \times$ Dijkstra $\to n^3$ (Dijkstra: n^2) $\quad n \times$ Bellman-Ford $\to n^4$ (Bellman-Ford: n^3)

Ukážeme: Floydův-Warshallův alg. také v n^3, ale triviální

Df: D^k_{ij} := délka nejkratšího sledu z i do j, $\underbrace{\text{jehož vnitřní vrcholy leží v } \{1 \ldots k\}}_{\text{„přes } 1 \ldots k\text{"}}$

↳ matice $D^0, D^1, \ldots, D^n$
D^0 = matice L, $D^n = D$

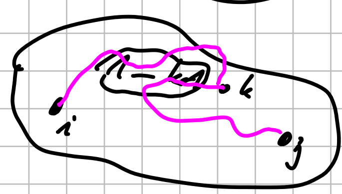


Výpočet D^k z D^{k-1}

$$D^k_{ij} \leftarrow \min\left(D^{k-1}_{ij},\ D^{k-1}_{ik} + D^{k-1}_{kj}\right)$$

D^{k-1}_{ij}: k nepoužito
$D^{k-1}_{ik} + D^{k-1}_{kj}$: k použito (právě 1×)

čas $\Theta(n^2)$

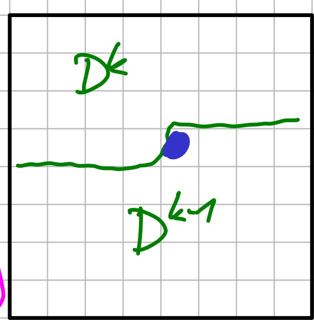


n kroků: $D^0 \to D^1 \to D^2 \to \ldots \to D^n$ čas $\Theta(n^3)$

Nevýhoda: Paměť $\Theta(n^3)$ ⟶ stačí si pamatovat D^{k-1} a D^k prostor $\Theta(n^2)$

↳ přepisování na místě

👁 $D^{k-1}_{ik} = D^k_{ik}$
$D^{k-1}_{kj} = D^k_{kj}$
} přepis neškodí (nic nezmění)



Celé: Pro $k = 1, \ldots, n$:
  Pro $i = 1, \ldots, n$:
    Pro $j = 1, \ldots, n$:
      $D_{ij} = \min(D_{ij},\ D_{ik} + D_{kj})$

čas $\Theta(n^3)$, prostor $\Theta(n^2)$