Tomáš Oberhuber

Faculty of Nuclear Sciences and Physical Engineering
Czech Technical University in Prague

Video na Youtube

# Třídění pomocí reversí

- budeme řešit úlohu řazení pomocí reversí – *sorting by reversals*
- jde o úlohu, jak převést permutaci

$$\pi = \pi_1 \pi_2 \dots \pi_n$$

na identickou permutaci

$$\sigma = 12 \dots n.$$

- omezíme se přitom jen na určitý typ operací nad permutacemi, tzv. reverse

# Třídění pomocí reversí

## Definition 1

Reverse $\rho(i,j)$ je taková permutace, která převrátí část permutace $\pi$ mezi pozicemi $i$ a $j$, tj.

$$\rho(i,j)\pi = \pi_1 \dots \pi_{i-1}\pi_j\pi_{j-1} \dots \pi_{i+1}\pi_i\pi_{j+1} \dots \pi_n.$$

## Definition 2

Reversní vzdálenost mezi dvěma permutacemi $\pi$ a $\sigma$ je minimální počet reversí, kterými lze převést $\pi$ na $\sigma$, značíme $d(\pi,\sigma)$. $\sigma$ obecně nemusí být identická permutace.

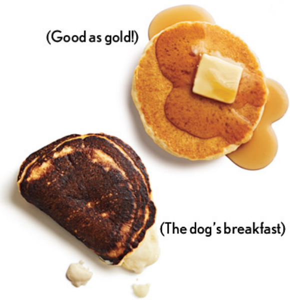
# Třídění pomocí reversí

- úloha třídění pomocí reversí znamená najít právě onu nejkratší posloupnost, která převede libovolnou permutaci $\pi$ na identickou

- podobná úloha je známá jako Třídění palačinek – *Pancake flipping problem*
    - kuchař hází palačinky různých průměrů na talíř, číšník je pak musí srovnat od největší k nejmenší a smí vždy vzít jen několik palačinek **z vrchu** a otočit je

# Třídění pomocí reversí

- složitější úloha je Třídění spálených palačinek – *Burnt pancake flipping problem*
  - každá palačinka je zespodu spálená, na konci třídění nesmí být žádná spálenou stranou nahoru



# Třídění pomocí reversí

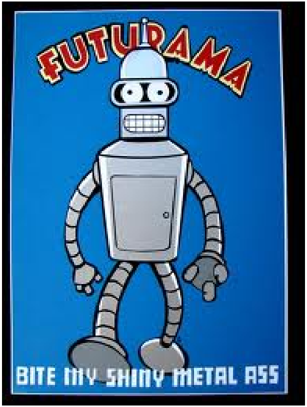
Hladový algoritmus

- v $i$-tém kroku uděláme takovou reversi, že $i$-tý prvek permutace se dostane na svou cílovou pozici
- tj. úlohu lze vyřešit po maximálně n krocích
- u palačinkového třídění je to $2n$
    - každou palačinku si musím nejdřív otočit na vrch, a pak na své místo

# Třídění pomocí reversí

- Gates W. H., Papadimitriou Ch. H., *Bounds for sorting by prefix reversal*, Discrete Mathematics, 27 (1979), pp. 47–57.
- ukázali, že je $17n/16 \leq f(n) \leq (5n+5)/3$ (pro palačinkové třídění)

# Třídění pomocí reversí




- Cohen D. S., Blum M., *On the problem of sorting burnt pancakes*, Discrete Applied Mathematics, 61 (1995), pp. 105–120.
- ukázali, že je $3n/2 \leq g(n) \leq 2n - 2$

# Třídění pomocí reversí

- lepší výsledek se zatím nezná
- Caprara A., *Sorting by reversals is difficult*, Proceedings of the first annual international conference on Computational molecular biology, pp. 75–83, 1997.
- úloha je NP-úplná
- ukážeme si aproximační algoritmus, který dává jen přibližně nejlepší řešení

## Definition 3

Buď $A(\pi)$ počet kroků získaných aproximačním algoritmem a $Opt(\pi)$ počet kroků optimálního řešení pro danou vstupní permutaci $\pi$. Pak aproximační poměr algoritmu je definován jako

$$\max_{|\pi|=n} \frac{A(\pi)}{Opt(\pi)}.$$

# Třídění pomocí reversí

- Elias I., Hartman T. *A 1.375-Approximation Algorithm for Sorting by Transpositions*, IEEE/ASM Transactions on Computational Biology and Bioinformatics, 3(4), pp.369–379, 2006.
- my si ukážeme algoritmus s aproximačním poměrem 4

# Třídění pomocí reversí

Mějme libovolnou permutaci $\pi$ délky $n$, tj.

$$\pi = \pi_1, \ldots \pi_n.$$

Provedeme její rozšíření na

$$\pi_0 \pi_1 \ldots \pi_n \pi_{n+1} = 0 \pi_1 \ldots \pi_n (n+1).$$

## Definition 4

Dvojici prvků $\pi_i, \pi_{i+1}$ nazveme sousedními, pokud platí, že $\pi_i = \pi_{i+1} - 1$ nebo $\pi_i = \pi_{i+1} + 1$. Pokud ne, jde o bod zlomu (*breakpoint*). Počet bodů zlomu v permutaci $\pi$ budeme značit jako $b(\pi)$.

# Třídění pomocí reversí

## Definition 5

*Strip* je interval v permutaci mezi dvěma sousedícími body zlomu. Rozlišujeme rostoucí a klesající strip. Jednoprvkové stripy definujeme jako klesající kromě prvního (0) a posledního ($n+1$), které definujeme jako rostoucí.

## Remark 6

$$\underrightarrow{0}, \underleftarrow{2, 1}, \underrightarrow{3, 4, 5}, \underleftarrow{8, 7, 6}, \underrightarrow{9}$$

# Třídění pomocí reversí

## Remark 7

- *vhodná reverse dokáže odstranit jeden nebo dva breakpointy*
- *pokud dokážeme, že nové breakpointy nevznikají, můžeme sestrojit algoritmus pro jejich minimalizaci, až dostaneme jeden rostoucí strip bez breakpointů*

# Třídění pomocí reversí

## Theorem 8

*Pokud permutace $\pi$ obsahuje klesající strip, pak existuje reverse $\rho(i, j)$, která sníží počet breakpointů, tj.*

$$b(\rho\pi) < b(\pi).$$

## Proof.

- ze všech klesajících stripů najdeme ten, který obsahuje nejmenší element $k$
- element $k - 1$ tedy musí náležet do rostoucího stripu a zároveň ho musí zakončovat
- $k$ a $k - 1$ patří ke dvěma různým breakpointům
- otočením segmentu mezi prvky $k$ a $k - 1$ spojíme rostoucí strip končící prvkem $k - 1$ s původně klesajícím stripem končícím prvkem $k$ do jednoho stripu
- tím jsme zrušili jeden breakpoint

# Třídění pomocí reversí

## Remark 9

*Pokud permutace neobsahuje klesající strip, např.*

$$\underrightarrow{0,1},\underrightarrow{5,6,7},\underrightarrow{2,3,4},\underrightarrow{8,9}$$

*tak jeden rostoucí strip otočíme.*

# Třídění pomocí reversí

**procedure** BREAKPOINTREVERSALSORT( $\pi$ )
    **while** $b(\pi) > 0$ **do**
        **if** $\pi$ má klesající strip **then**
            najdi reversi $\rho$, která minimalizuje $b(\rho\pi)$
        **else**
            najdi reversi $\rho$, která otočí některý strip na klesající
        **end if**
        $\pi := \rho\pi$
        vypiš $\pi$
    **end while**
**end procedure**

# Třídění pomocí reversí

Příklad:

$$\pi = 8, 2, 7, 6, 5, 1, 4, 3$$

# Třídění pomocí reversí

## Theorem 10

*Zmíněný algoritmus (BreakPointReversalSort) je aproximační algoritmus s poměrem maximálně 4.*

## Proof.

- žádná reverse nemůže snížit počet breakpointů o více než o dva, tj.
$$d(\pi) \geq \frac{b(\pi)}{2}$$
- algoritmus v nejhorším případě odstraní jeden breakpoint každé dva kroky
- $$\max_{|\pi|=n} \frac{A(\pi)}{Opt(\pi)} = \max_{|\pi|=n} \frac{2b(\pi)}{d(\pi)} \leq \max_{|\pi|=n} \frac{2b(\pi)}{b(\pi)/2} = 4$$

□