13.hodina 4.12. 2023

pondělí 4. prosince 2023 11:53

Radix sort (řazení po řádech)

- původ = Herman Hollerith (1887) strojové zpracování výsledků sčítání lidu 1890 - USA
  => předchůdce IBM
- popis - používá nepřímé porovnání, většinou stabilní
  - řazení po řádech - symbolů s různou vahou

řád 1.	řád 0.
3	8
1	2
2	5

=> nejčastěji celá čísla

=> lze řadit libovolné řetězce

"125" (řády 2. 1. 0.) "1001" (řády 3. 2. 1. 0.) "Ahoj" (řády 3. 2. 1. 0.) - řetězec

$$1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0$$

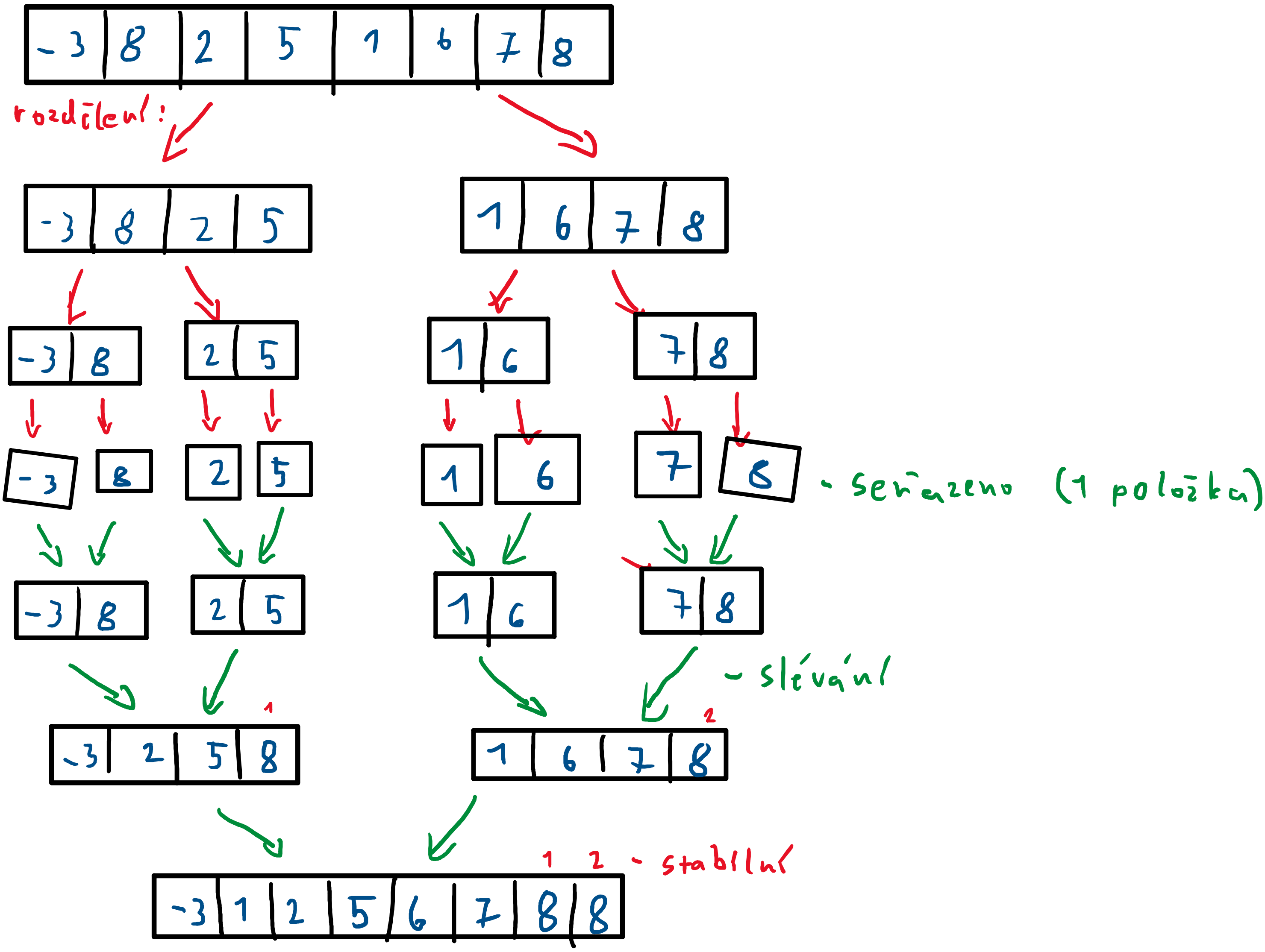
- **analýza** - složitost $O(n) \Rightarrow O(k \cdot C(n))$
  - k = počet řádů
  - $C(n)$ = složitost 1 průchodu $\Rightarrow O(n)$
  - $C(n) = C(n+m)$
    - m: vytvoření výstupního pole
    - n: čtení všech položek
- **výhoda**: - nejvýkonnější, vždy $O(n)$
- **nevýhoda**: - paměťová složitost $O(n)$

Merge sort - řazení sléváním

- **původ** - John Von Neumann (1945)
- **popis** - stabilní řadící algoritmus
  - metoda rozděl a panuj => vhodná pro paralelizaci
  - složitost: - výpočetní $O(n \cdot \log n)$
    - paměťová $O(1)$
- **implementace** - pro pole a seznam
  - 2 varianty: - top-down => rekurze
    - bottom-up => iterace pomocí cyklů
- **princip** - slévání seřazených polí
  - => výsledkem = seřazené pole
  - => výp. složitost slévání je $O(n)$
  - v ideálním případě provedeno $\log_2 n$ slévání
- **fáze**:
  1. rozdělit neseřazenou část na 2 stejně velké části
  2. jedna položka je již seřazená - konec rekurze
  3. slévání seřazených částí

15.hodina 18.12. 2023

pondělí 18. prosince 2023 12:14



- na začátku slévání jsou slévána kratší pole => rychlé
- výhoda : - stabilní
  - lze paralelizovat
  - vhodné pro sekvenční paměti -> HDD
  - lze optimalizovat pro menší paměti
- nevýhody : - potřeba zásobníku
- využití : - databázové indexy

Quick sort (řazení dělením)

- původ = Tony Hoare (1960) => program pro překlad (řazení slov)
- popis: - nestabilní (pouze 1 průchod)
  - in-place => O(1) paměti
  - rozděl a panuj => výp. složitost $O(n \cdot \log n)$
  - paralelizace
- princip: - využití <u>pivota</u> => prvek z řazeného pole podle kterého jsou položky porovnávány
  1. fáze: - výběr pivota => většinou první položka neseř. pole
     - ideální pivot je uprostřed seřazeného pole
       => nejrychlejší řešení
       => při nevhodném pivotovi -> až složitost $O(n^2)$
  2. fáze: - prvky < pivot -> přesunuty před pivota
     - prvky > pivot -> přesunuty za pivota
     => rozdělení
  3. fáze: - neseřazené části -> řazeny stejným postupem
     => panuj
     => řazení končí když neseřazená část je n = 1
- analýza: - výběr pivota => O(1)
  - umístění pivota => O(n)