Třídíci algoritmy Bubble Sort

xplagi0b

1. března 2020

Třídící (řadící) algoritmus

- Třídící algoritmus zajišťuje uspořádání daných dat podle požadovaného pořadí
- Řadí na základě určené části dat ("klíče")
- Nejčastěji jde o numerické či abecední seřazení

Třídící (řadící) algoritmus

- Třídící algoritmus zajišťuje uspořádání daných dat podle požadovaného pořadí
- Řadí na základě určené části dat ("klíče")
- Nejčastěji jde o numerické či abecední seřazení
- Na vstupu algoritmu je posloupnost záznamů:

$$S = S_1, S_2, \ldots, S_n$$

Třídící (řadící) algoritmus

- Třídící algoritmus zajišťuje uspořádání daných dat podle požadovaného pořadí
- Řadí na základě určené části dat ("klíče")
- Nejčastěji jde o numerické či abecední seřazení
- Na vstupu algoritmu je posloupnost záznamů:

$$S = S_1, S_2, \ldots, S_n$$

- A výstupem je:
 - Posloupnost je seřazená:

$$S_1' \leq S_2' \leq \cdots \leq S_n'$$

 Posloupnost je permutací původní posloupnosti S (pořadí se mohlo změnit, avšak data zůstavají stejná)



Složitost algoritmu

- Neexistuje třídící algoritmus dokonalý pro všechna použití
- Pro kokrétní použití se volí vhodný algoritmus na základě jeho vlastností, především:
 - Časová složitost
 - Prostorová složitost
 - Náročnost implementace
 - ...

- též řazení záměnou je algoritmus opakovaně procházející seznam, kde při každém projití porovnává sousedící prvky a případě potřeby je prohodí
- implementačně jednochý algoritmus s časovou náročností O(n²), kvůli které se v praxi příliš nepoužívá

- též řazení záměnou je algoritmus opakovaně procházející seznam, kde při každém projití porovnává sousedící prvky a případě potřeby je prohodí
- implementačně jednochý algoritmus s časovou náročností O(n²), kvůli které se v praxi příliš nepoužívá
- Postup algoritmu pro 3 prvky:

8 4 2 Nalezeno špatné pořadí 8 a 4, prohazuje

- též řazení záměnou je algoritmus opakovaně procházející seznam, kde při každém projití porovnává sousedící prvky a případě potřeby je prohodí
- implementačně jednochý algoritmus s časovou náročností O(n²), kvůli které se v praxi příliš nepoužívá
- Postup algoritmu pro 3 prvky:

8 4 2 Nalezeno špatné pořadí 8 a 4, prohazuje

4 8 2 Nalezeno špatné pořadí 8 a 2, prohazuje

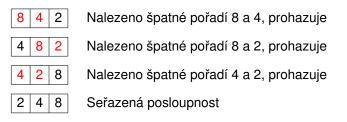
- též řazení záměnou je algoritmus opakovaně procházející seznam, kde při každém projití porovnává sousedící prvky a případě potřeby je prohodí
- implementačně jednochý algoritmus s časovou náročností O(n²), kvůli které se v praxi příliš nepoužívá
- Postup algoritmu pro 3 prvky:

8	4	2	Nalezeno špatné pořadí 8 a 4, prohazuje
---	---	---	---

4 8 2 Nalezeno špatné pořadí 8 a 2, prohaz	zuje
--	------

4 2 8 Nalezeno špatné pořadí 4 a 2, prohazuje

- též řazení záměnou je algoritmus opakovaně procházející seznam, kde při každém projití porovnává sousedící prvky a případě potřeby je prohodí
- implementačně jednochý algoritmus s časovou náročností O(n²), kvůli které se v praxi příliš nepoužívá
- Postup algoritmu pro 3 prvky:



Příklad implementace

```
//array[n], n - number of items
for(int i = 0; i < n-1; i++)
for(j = 0; j < n-i-1; j++)
if(array[j] > array[j+1]) {
    int temp = array[j];
    array[j] = array[j+1];
    array[j+1] = temp;
}
```

Zdroje



Wikipedie, P.: Bublinkové řazení. [online], rev. 24.11.2018, [vid. 05.05.2019]. Dostupné z:

https://cs.wikipedia.org/w/index.php?title= Bublinkov%C3%A9_%C5%99azen%C3%AD&oldid=16711712



Wikipedie, P.: Řadící algoritmus. [online], rev. 30.10.2018, [vid. 05.05.2019]. Dostupné z:

https://cs.wikipedia.org/w/index.php?title= Bublinkov%C3%A9_%C5%99azen%C3%AD&oldid=16711712