

Třídění

Úloha

- Chceme setřídít data uložená v poli / spojovém seznamu / souboru
- Typy třídění:
 - Vnitřní – data uložená ve vnitřní paměti PC
 - Vnější – data jsou při třídění ukládána na disk (celá se do paměti nevejdou)

Přímé metody třídění

- Krátké jednoduché algoritmy
- Třídí tzv. na místě – přímo v poli, používají jen konstantní pomocnou paměť (pomocné proměnné)
- Typická časová složitost $O(n^2)$
- Vhodné, pokud třídíme pouze málo dat

Typičtí představitelé

- Selection sort
- Insertion sort
 - Binary insertion sort
- Bubble sort
- Shaker sort
- Shell sort

Selection sort – přímý výběr minima

- Základní myšlenka

1. Najdeme v poli nejmenší prvek a prohodíme jej s prvkem na indexu 0
2. Najdeme ve zbytku (od indexu 1) nejmenší prvek a prohodíme s prvkem na indexu 1

...

1. Najdeme ve zbytku pole (od indexu i) nejmenší prvek a prohodíme s prvkem na indexu i , $\forall i \in \{2, 3, \dots, n - 2\}$

- Varianty:

- Naivní – prohazují prvky pokaždé, kdy narazím na menší
- Lepší – pamatují si index minima a prohodím prvky jen jednou

Selection sort – přímý výběr minima

- Časová složitost
- $O(n^2)$
- Prostorová složitost
- $O(1)$

Insertion sort – třídění vkládáním

- Myšlenka – postupné vytváření setříděného pole zleva doprava
- Pro $i = 1, \dots, n - 1$ pole až do prvku s indexem $i - 1$ je už setříděná. Prvek na indexu i vezmeme a vložíme na správné místo.
- Časová složitost: $O(n^2)$
- Prostorová složitost: $O(1)$

Bubble sort

- Základní naivní varianta:
- $(n - 1)$ krát probubláme celé pole zleva doprava:
 - Pro $i = 0, \dots, n - 2$ porovnáme prvky na indexech i a $i + 1$, pokud je prvek na indexu $(i + 1)$ menší, prvky prohodíme
- Vylepšení:
 1. Pokud se v posledním probublávání neprohodily žádné prvky \rightarrow pole je již setříděné
 2. Probubláváme celým polem pouze do indexu $n - i - 2$, dál je pole již setříděné
 3. Neprobubláváme celým polem. V každém probublávání (průchodu) si zapamatujeme index nejpravějšího prvku, který se prohodil. V dalším průchodu stačí probublávat jen do tohoto indexu (-1) .

Shaker sort – třídění přetřásáním

- Proubláváme polem střídavě zleva a zprava, průběžně aktualizujeme levou a pravou stranu viz minulé vylepšení bubble sortu.
- Složitost bubble sortu a shaker sortu:
- Časová složitost: $O(n^2)$
- Prostorová složitost: $O(1)$