

Algoritmy řazení $O(N \log N)$

Z FAV wiki

// přidat ukázky

Společné pro všechny: Řadíme stromem.

Obsah

- 1 Řazení haldou (heapsort)
- 2 Shellovo řazení (shellsort)
- 3 Řazení dělením (splitsort, quicksort)
- 4 Řazení slucováním (mergesort)

Řazení haldou (heapsort)

Na začátku je neseřazené pole. Postupně obnovujeme nad tímto polem haldu pro první dva, tři, čtyři atd. prvky, až máme na poli vytvořenu haldu. Následně zaměníme nejvyšší prvek (vyjmeme ho) s posledním a obnovíme vlastnost haldy na $n-1$ prvcích. Obnovení haldy je $O(\log n)$ a obnovujeme po každém výběru, tedy $O(n \log^2 n)$. To opakujeme až zbude jednoprvková halda a celé pole je seřazené. Nestabilní (složitost závisí na vstupních datech).

V kostce: na poli uděláme haldu a max prvky vyměňujeme s posledními prvky haldy a haldu obnovujeme.

Shellovo řazení (shellsort)

Posloupnost rozdělíme na podposloupnosti s krokem h (každý h -tý počínaje prvním, počínaje druhým až počínaje $h-1$ -ním), které nezávisle seradíme vkládáním (insertsortem). Následně snižujeme hodnotu h a provádíme totéž až do serazení s $h=1$. Prvky daleko od své výsledné pozice se k ní dostanou v menším počtu kroků, než u obyčejného insertsortu.

- Shell - počáteční h = polovina délky pole, snížení vydělením dvěma
- Knuth - posloupnost $3i+1$, počáteční h = její prvek v nejbližší třetině délky pole

Jednoduchý a efektivní algoritmus, složitá analýza, nestabilní.

V kostce: rozdělení do stromu podle kroku (třeba na osminy), insertsort větví, a pak insertsort čtvrtin, polovin, konec

Poměrně zajímavě udělaná ukázka z kartama: <http://www.youtube.com/watch?v=QG8hs0wqmqk>

Řazení dělením (splitsort, quicksort)


Určíme prvek, jehož hodnota bude rozdělovat pole (pivot). Procházíme z obou stran pole a menší hodnoty vpravo prohazujeme s většími vlevo, až se oba průchody potkají. Na tuto pozici přesuneme pivot - vlevo jsou hodnoty menší, vpravo větší. Totéž provedeme s částí pole vlevo od pivotu i vpravo od něj. Rekurze končí, je-li předán pouze jeden prvek k seřazení. Nejčasteji se pivot určuje jako krajní prvek pole (nezasahuje pak do procesu dělení). Lze použít zásobník místo rekurze. Snadno implementovatelné, nestabilní.

V kostce: rozdělení do stromu podle pivotů (prvek velikostně cca uprostřed podstromu, ideálně medián, ale může být i náhodný), v každém podstromu vzájemně vyměníme větší a menší prvky než pivot.

Výukové videjko: http://www.youtube.com/watch?v=y_G9BkAm6B8

Řazení slučováním (mergesort)

Pole je rekurzivně rozdělováno na poloviny. Dojdeme až na úroveň dvojic prvku, které seřadíme (sloučíme). O úroveň výše opět po dvojicích sloučíme (kopírujeme vždy menší prvek z obou polí a posuneme se za něj) a postupujeme až sloučíme obě poloviny původního pole. Je to nejčasteji používaná metoda a je stabilní.)

V kostce: jako když mícháme karty - 2 balíčky a děláme  ???????

. Procházíme 2 podstromy a řadíme posunem 2 indexů, jeden v každém podstromu, a vždy zapíšeme větší prvek pod jedním z indexů, který posuneme

Bezva video - <http://www.youtube.com/watch?v=GCae1WNvnZM>

Nestabilní metody lze "zestabilnit" například zpřeházením dat na vstupu.

Radix sort =???

Citováno z „[http://www.512.cz/index.php?title=Algoritmy_%C5%99azen%C3%AD_O\(N_logN\)“](http://www.512.cz/index.php?title=Algoritmy_%C5%99azen%C3%AD_O(N_logN)“)

Kategorie: Fav-kiv-bzinf

- Stránka byla naposledy editována 20. 2. 2014 v 06:36.
- Stránka byla zobrazena 735krát.