Třídění

Jak se liší stabilní a nestabilní třídící algoritmy?

Heap-sort - jak zkonstruovat haldu všech prvků v lineárním čase?

Co je složitost v průmerném případě?

Porovnejte Quick-sort a Heap-sort

Navrhněme algoritmus lepší než oba algoritmy, stačí je vhodně zkřížit a analyzovat.

Co jsou Bucket-sort, Counting-sort a Radix-sort? Jakých vlastností vstupu a podrocedur využívají?

(Podle času Bubble-sort jako hradlová síť, bitonické třídění)

Master theorem

Pro rekurzi $T(n) = aT(n/b) + f(n)$ definujme $c = log_b(a)$
Předpokládáme, že a, b, k jsou rozumné konstanty
Pokud $f(n) \in O(n^c)$ potom $T(n) \in \Theta(n^c)$.
Pokud $f(n) \in \Theta(n^c * log^k n)$, potom $T(n) \in \Theta(n^c * log^{k+1} n)$.
Pokud $f(n) \in \Omega(n^c)$, potom $T(n) \in \Theta(f(n))$
(existuje spoustu dalších rozšíření)

Porovnejte

1. $T(n) = 2T(n/3) + \mathcal{O}(n)$
2. $T(n) = 2T(n/3) + \Theta(n)$
3. $T(n) = 3T(n/2) + \Theta(n)$ (násobení čísel)
4. $T(n) = 2T(n/2) + \Theta(n)$ (merge sort)
5. $T(n) = 2T(n/2) + \Theta(n^2)$ (body v rovině)
6. $T(n) = 7T(n/2) + \Theta(n^2)$ (Strassen)
7. $T(n) = T(n/2) + \Theta(1)$ (bin. search)
8. $T(n) = T(n/2) + T(n/3) + \Theta(n)$
9. $T(n) = T(7n/10) + T(n/5) + \Theta(n)$ (quick select)
10. $T(n) = 2T(n) + \Theta(nlogn)$
11. $T(n) = n^{1/2}T(n^{1/2}) + \Theta(n)$ (e.g. exp. stromy)
12. $T(n, m) = T(n/4, x) + T(n/4, m - x) + \Theta(n + m)$ (KKT MST)

Domácí úkol

První část

Pomocí metody rozděl a panuj ($D\&C$) se dají dlouhá čísla A a B (nebo matice) násobit efektivněji než klasickým přístupem. Nabízí se otázka, zda by nešlo využít speciálních vztahů mezi A a B.

Mohlo by třeba platit, že když $A = B$ (tedy počítáme A^2), dá se výsledek spočítat rychleji než obecné násobení.

Ukažte, že nikoliv. Ukažte, že problém výpočtu druhé mocniny čísla X (délky n) je asymptoticky alespoň tak složitý jako násobení dvou různých čísel A a B (stejné délky n). Chceme vlastně ukázat, že pomocí konstantně mnoho mocnění (délky zhruba n) a jednodušších operací umíme simulovat výpočet obecného násobení. Nezapomeňte správně asymptoticky analyzovat složitost postupu.

Hint: Stejně jako u $D\&C$ přístupu vyžijeme toho, že umíme sčítat a odčítat linárně a tedy určitě asymptoticky rychleji, než násobit.

Druhá část

Vyřešte a porovnejte

a. $T(n) = 2T(n/4) + \Theta(\sqrt{n})$

b. $T(n) = 4T(n/2) + \Theta(nlog(n))$

c. $T(n) = T(n/2) + T(n) + \Theta(log(n))$

d. $T(n) = 3T(n/3) + \Theta(nlog(n))$

Bonusová část

Zkuste zkrotit rekurzi

$T(n) = 3T(n/2) + 4T(n/4) + \Theta(n)$

Master theorem je na takovou rekurzi krátký. Plné hodnocení bude i za dobře zdůvodněné a vzájemně blízké spodní a horní odhady. Existuje několik způsobů jak se poměrně nepracně dostat na přesnost o zhruba jednom desetinném místě, které využívají pouze myšlenky Master theoremu, substituční metody, a prosté hrubosti odhadu. Dejte pozor, aby vaše závěry byly korektní.