Ugeseddel: Analyse af algoritmer

Philip Bille

Om denne uge

Litteratur *Introduction to Algorithms*, Cormen, Rivest, Leisersons og Stein (CLRS): Kap 3.

Opgaver

1 $[o]$ **Relativ vækst** Arranger følgende funktioner i voksende rækkefølge efter asymptotisk vækst. Dvs. hvis funktionen $g(n)$ følger umiddelbart efter funktionen $f(n)$ i din liste skal der gælde at $f(n) = O(g(n))$.

$$n \log n \qquad n^2 \qquad 2^n \qquad n^3 \qquad \sqrt{n} \qquad n$$

2 **Θ-notation** Skriv følgende udtryk med Θ-notation.

$$n^2 + n^3/2$$
$$2^n + n^4$$
$$\log_2 n + n\sqrt{n}$$
$$n(n-6)$$
$$4\sqrt{n}$$
$$8\log_2^7 n + 34\log_2 n + \frac{1}{1000} n$$
$$2^n 7 + 5\log_2^3 n$$
$$n(n^2 - 18)\log_2 n$$
$$n\log_2^4 n + n^2$$
$$n^3 \log_2 n + \sqrt{n}\log_2 n$$

3 **Løkkelige løkker** Analyser køretiden af følgende løkker som funktion af n og udtryk resultatet i Θ-notation.

Algoritme 1 Løkke1(n)

$i = 1$
while $i \leq n$ **do**
  print "$\star$"
  $i = 2 \cdot i$
end while

Algoritme 2 Løkke2(n)

$i = 1$
while $i \leq n$ **do**
  print "$\star$"
  $i = 5 \cdot i$
end while

Algoritme 3 Løkke3(n)

```
for i = 1 to n do
    j = 1
    while j ≤ n do
        print "⋆"
        j = 2 · j
    end while
end for
```

4 Asymptotiske påstande Hvilke af følgende påstande er korrekte?

$$\frac{1}{20}n^2 + 100n^3 = O(n^2)$$

$$\log_2 n + n = O(n)$$

$$2^{\log_2 n} = O(n)$$

$$n^3(n-1)/5 = \Theta(n^3)$$

$$\log_2^2 n + n = \Theta(n)$$

$$\frac{n^3}{1000} + n + 100 = \Omega(n^2)$$

$$2^n + n^2 = \Omega(n)$$

$$\log_4 n + \log_{16} n = \Theta(\log n)$$

$$n^{1/4} + n^2 = \Theta(n)$$

$$2^{\log_4 n} = \Theta(\sqrt{n})$$

5 Fordoblinghypoteser Løs følgende opgaver.

5.1 [o] Algoritme A kører i præcis $7n^3$ tid på input af størrelse n. Hvor meget langsommere kører algoritmen hvis du fordobler inputstørrelsen?

5.2 [D] Algoritme B kører på input af størrelsen 1000, 2000, 3000, 4000 og 5000, i henholdsvis 5, 20, 45, 80, og 125 sekunder. Estimer hvor lang tid det vil tage at køre B på et input af størrelse 6000. Hvad er køretiden af B udtrykt i Θ-notation?

5.3 Algoritme C kører 3 sekunder langsommere hver gang man fordobler størrelsen af input. Hvad er køretiden af C udtrykt i Θ-notation?

6 Asymptotiske egenskaber Løs følgende opgaver.

6.1 CLRS 3.1-1

6.2 CLRS 3.1-3

6.3 CLRS 3.1-4

6.4 [$C*$] Vis at $\log_2(n!) = \Theta(n \log n)$. *Hint:* Start med at vise den øvre grænse.

7 Skæve fletninger Professor Gørtz foreslår følgende variant af flettesortering kaldet 3-flettesortering. 3-flettesortering fungerer præcis som flettesortering pånær at man deler tabellen i 3 trejdedele, som sorteres rekursivt og flettes sammen. Løs følgende opgaver.

7.1 Vis at man kan flette 3 tabeller i lineær tid.

7.2 Analyser køretiden af 3-flettesortering.

7.3 [$*$] Generaliser algoritmen og analysen af 3-flettesortering til k-flettesortering for $k > 3$. Er k-flettesortering en forbedring af standard 2-flettesortering?

8 Maksimal deltabel Lad $A[0..n-1]$ være en tabel af heltal (både positive og negative). En *maksimal deltabel* af A er en deltabel $A[i..j]$ således at summen $A[i]+A[i+1]+\cdots+A[j]$ er maksimal blandt alle deltabeller af A. Løs følgende opgaver.

8.1 $[o]$ Giv en algoritme, der finder en maksimal deltabel i A i $O(n^3)$ tid.

8.2 Giv en algoritme, der finder en maksimal deltabel i A i $O(n^2)$ tid. *Hint:* Vis at man kan beregne summerne for alle deltabeller i $O(1)$ tid per deltabel.

8.3 $[**]$ Giv en del-og-hersk algoritme, der finder en maksimal deltabel i A i $O(n \log n)$ tid.

8.4 $[**]$ Giv en algoritme, der finder en maksimal deltabel i A i $O(n)$ tid.

9 Frivillig afleveringopgave: Mere relativ vækst Arranger følgende funktioner i voksende rækkefølge efter asymptotisk vækst. Dvs. hvis funktionen $g(n)$ følger umiddelbart efter funktionen $f(n)$ i din liste skal der gælde at $f(n)=O(g(n))$.

$$5000 \log_2 n \qquad \frac{n}{\log_2 n} \qquad \frac{1}{4}n^2 - 10000n \qquad n^{1/100} \qquad 4n \log_2 n \qquad \sqrt{n}+7 \qquad 8n$$