

Aflevering uge 2

Magnus, Mikkel og Sara

21. september 2020

Del 1 $f(x)$ er $O(g(x))$ hvis der kan findes $c > 0$ således at $f(x) \leq cg(x)$ for alle $x \geq x_0$

I dette tilfælde skal vi vise at $n^2 - 8n + 100 \log_2(n)$ er $O(n^2)$

Lad $c = 1$. Vi rykker herefter alle led, på nær logaritmen, over på venstre side af ulighedstegnet over på højre side. Vi får derfor $100 \log_2(n) \leq 8n$

Denne ulighed er sand for $x_0 = 79$ således at $x \geq x_0$ og derfor kan det konkluderes at $n^2 - 8n + 100 \log_2(n)$ er $O(n^2)$

Del 2 Vi benytter reglerne fra noterne til at vise om hvorvidt $f(n)$ er $o(g(n))$ eller $g(n)$ er $o(f(n))$. Det noteres at hele idéen med at analysere hvordan funktionerne vokser asymptotisk, er for at vælge hvilke algoritmer man bruger for store inputs. Der vil det altid være funktionen der vokser langsomt, som man vil benytte for store inputs. For små inputs er det dog anderledes.

(a) Vi viser først at $g(n) = n^3 - 50n$ er $\Theta(n^3)$

n er $o(n^3)$ ifølge (L4)

(L1) siger så at $1 \cdot n^3 + (-50) \cdot n$ er $\Theta(n^3)$

For $f(n) = \log_2(n)$ gælder det at $\log_2(n)$ er $o(n^3)$ ifølge (L3)

Vi kæder dem nu sammen med (M4) som siger, at hvis $\log_2(n)$ er $o(n^3)$ og $n^3 - 50n$ er $\Theta(n^3)$ så vil $\log_2(n)$ være $o(n^3)$.

Til store inputs benytter vi altså $f(n) = \log_2(n)$

(b) Ifølge (L2) vil 1 være $o(\log_2(n))$

(L1) siger derfor at $1 \cdot \log_2(n) + 100 \cdot 1$ is $\Theta(\log_2(n))$

$\log_2(n)$ er $o(\sqrt{n})$ ifølge (L3)

Vi kæder dem sammen med (M4), således vil $\log_2(n) + 100$ være $o(\sqrt{n})$

Til sidst benytter vi (M3) og får altså at $n \cdot (\log_2(n) + 100)$ er $o(n\sqrt{n})$ hvilket var de funktion vi startede med. Til store inputs benytter vi altså $g(n) = n \log_2(n) + 100n$

(c) I denne opgave vil vi gerne omskrive udtrykkene til noget der er nemmere at have med at gøre. Lad os først tage logaritmen i base 2 på begge funktioner og se hvad der sker.

$$\log_2(2^{\log_4(n)}) = \log_4(n) = \frac{\log_2(n)}{\log_2(4)} = \frac{\log_2(n)}{2} = \log_2(\sqrt{n}) = \frac{1}{2} \log_2(n)$$

$$\log_2\left(\frac{2^n}{n^{100}}\right) = \log_2(2^n) - \log_2(n^{100}) = n - 100 \log_2(n)$$

I de to udregninger har jeg benyttet reglerne fra kapitel 1 i noterne.

Lad os kigge på hvordan disse to funktioner vokser asymptotisk.

(M2) siger at $\frac{1}{2} \log_2(n)$ er $\Theta(\log_2(n))$

(L3) siger at $\log_2(n)$ er $o(n)$ derfor siger (M4) at $\frac{1}{2} \log_2(n)$ er $o(n)$ og (L1) siger at

$1 \cdot n + (-100) \cdot \log_2(n)$ er $\Theta(n)$

Vi husker at Θ er symmetrisk, derfor gælder det også at n er $\Theta(n - 100 \log_2(n))$. (M4) siger nu at hvis $\frac{1}{2} \log_2(n)$ er $o(n)$ og n er $\Theta(n - 100 \log_2(n))$ så vil $\frac{1}{2} \log_2(n)$ være $o(n - 100 \log_2(n))$

Til slut opløfter vi de to udtryk i 2, da vi tog logaritmen i base 2. Vi kan da konkludere, at hvis $\frac{1}{2} \log_2(n)$ er $o(n - 100 \log_2(n))$ så er $2^{\frac{1}{2} \log_2(n)}$ også $o(2^{n - 100 \log_2(n)})$, som var vores to funktioner i udgangspunktet, bare omskrevet. Til store inputs benytter vi altså $f(n) = 2^{\log_4(n)}$

Del 3 Her finder vi en general forskrift for de rekursive sekvenser

- (a) Vi ser hurtigt at a_n bare ændre sig fra 2 til 200 for hver gang n stiger med 1, således at

$$f_a(n) = \begin{cases} 2, & \text{for } n = \text{odd.} \\ 200, & \text{for } n = \text{even.} \end{cases} \quad (1)$$

Vi ser at der lægges 2 til hver gang n stiger med 1, derfor er

$$f_b(n) = 2n$$

Vi ser at den forrige værdi ganges på, helt ned til 1, hvilket ligner fakulteten af n

$$f_c(n) = n!$$

- (b) $f_a(n)$ er $\Theta(1)$, da det effektivt bare er en konstant, dette følger af (L6). Der er ingen stigning. Dette er altså den laveste størrelsesorden af de 3.

$f_b(n)$ er $\Theta(n)$, det følger af (M2). Dette er den midterste størrelsesorden.

$f_d(n)$ er $\Theta(n \log_2(n))$, det følger af theorem 15 i noterne. Dette er altså funktionen der gror hurtigst.

Ifølge (L2) vil $n^0 = 1$ være $o(\log_2(n))$. Det følger derfor af (M3) at n er $o(n \log_2(n))$.

(L4) siger at $n^0 = 1$ er $o(n)$, derfor siger (M1) at $n^0 = 1$ er $o(n \log_2(n))$

Der er derfor redegjort for at $f_a(n)$ er $o(f_b(n))$ som er $o(f_d(n))$ er den rigtige rækkefølge.

Del 4 Her er det explicitte udtryk bestemt. Theorem og example henviser til noterne.

Fra theorem 14 ser vi at vi kan dele summen op

$$\sum_{k=0}^n (2k+1) = \sum_{k=0}^n 2k + \sum_{k=0}^n 1$$

Theorem 14 fortæller også at

$$\sum_{k=0}^n 2k = 2 \sum_{k=0}^n k$$

Fra theorem 13 ser vi at

$$\sum_{k=1}^n k = \frac{n^2 + n}{2}$$

Det noteres at forskellen fra $k=0$ til $k=1$ er 0, da $2 \cdot 0 = 0$. Vi ser derfor at

$$2 \sum_{k=0}^n k = n^2 + n$$

Fra example 12 ved vi at

$$\sum_{k=0}^n 1 = n + 1$$

Da vi husker at $k=0$ også har en skal lægges til.

Det ses altså at

$$\sum_{k=0}^n (2k+1) = n^2 + n + (n+1) = n^2 + 2n + 1$$