DMat-07

3.3: Kompleksitet af algoritme.

n : mål for størrelsen af input.

$f(n)$: det største antal skridt algoritmen bruger hvis inputtet har størrelse n. (worst case)

Find et simpelt udtryk $g(n)$ så $f(n)$ er $O(g(n))$.

Vi siger at algoritmen har (tids-) kompleksitet $O(g(n))$.

Algoritme 1: matrix multiplikation ($m = n = k$) Side 223

procedure $matrix\ multiplication(A,B$: $n \times n$ matricer)

for $i := 1$ to n

  for $j := 1$ to n

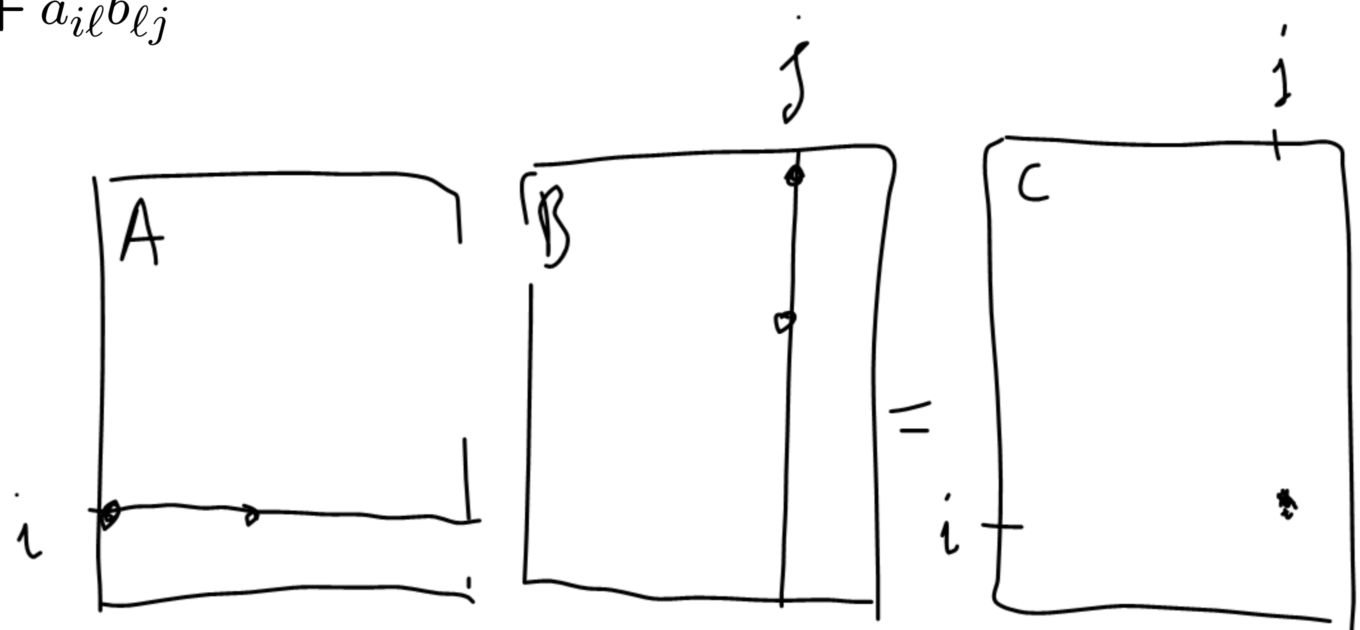
    $c_{ij} = 0$

    for $\ell := 1$ to n

      $c_{ij} := c_{ij} + a_{i\ell} b_{\ell j}$

return C

$\{\ C = AB\}$



$c_{ij} := 0$ udføres n^2 gange

$c_{ij} := c_{ij} + \underline{a_{il}} b_{lj}$ udføres n^3 gange

Kompleksitet: $O(n^3)$

procedure $insertion\ sort(a_1, \ldots, a_n$: reelle tal med $n \geq 2)$

for $j := 2$ **to** n

    $i := 1$

    while $a_j > a_i$

        $i := i + 1$

    $m := a_j$

    for $k := 0$ **to** $j - i - 1$

        $a_{j-k} := a_{j-k-1}$

    $a_i := m$

$\{a_1, \ldots, a_n$ er nu i voksende rækkefølge.$\}$

$a_1 \ldots a_i \ldots a_{j-1}\ a_j \ldots a_n$

For hvert j:

$a_j > a_i$ udføres i gange

sidste værdi af i

$a_{j-k} := a_{j-k-1}$ udføres $j-i$ gange

$a_i := m$ 1 gang

i alt: $i + j - i + 1 = j+1$ operationer

Samlede antal operationer

$$\sum_{j=2}^{n} j+1 = 3+4+5+\dots+(n+1) = 1+2+3+\dots+(n+1) - 3$$

$$\frac{(n+1)(n+1+1)}{2} - 3 = \frac{n^2+3n+2}{2} - 3$$

Kompleksitet: $O(n^2)$

En given computer kan udføre én operation på 10^{-9} sekund (et nanosekund).

Algoritme A bruger n^3 operationer for at løse et problem med input-størrelse n.

Algoritme B bruger $n^2 \log n$ operationer for at løse et problem med input-størrelse n.

Hvor stort et problem kan løses på et sekund af hver af de to algoritmer.

På 1 sekund: 10^9 operationer

Algoritme A: Find n så $n^3 = 10^9$

$$n = 10^3 = 1000$$

Algoritme B: Find n så $n^2 \log n = 10^9$

Mere præcis største $n \in \mathbb{Z}$: $n^2 \log n \leq 10^9$

Hvis $\log n = 10$: $n^2 \cdot 10 = 10^9 \Rightarrow n^2 = 10^8 \Rightarrow n = 10^4$

Løsning 8739

4.1: Hele tal: divisibilitet og modulær aritmetik.

$a, b \in \mathbb{Z}$, $a \neq 0$.
Vi siger at a går op i b, skrives $a \mid b$,
hvis der findes $c \in \mathbb{Z}$ så $b = ac$

$6 \mid 24$ da $24 = 6 \cdot 4$

$6 \mid 0$ da $0 = 6 \cdot 0$

$1 \mid n$ da $n = 1 \cdot n$

$a, b, c \in \mathbb{Z}$
$a \mid b \ \wedge \ a \mid c \ \Rightarrow \ a \mid b + c$,
$a \mid b \ \Rightarrow \ a \mid bc$, for alle heltal c.
$a \mid b \ \wedge \ b \mid c \ \Rightarrow \ a \mid c$.

$a \mid b \ \wedge \ b \mid c$ betyder der findes $p, q \in \mathbb{Z}$
så $b = a \cdot p$, $c = b \cdot q$

$$c = b \cdot q = a \cdot p \cdot q = a \cdot (p \cdot q)$$

Allbă $\quad a \mid c$

Division med rest.

$a, d \in \mathbb{Z}$, $d \geq 1$.
Der findes entydige tal $q, r \in \mathbb{Z}$ så

$$a = dq + r, \quad 0 \leq r < d.$$

Vi skriver: $r = a \mod d$.

Hvis $d \geq 1$: $\quad d \mid a \;\Leftrightarrow\; a \mod d = 0$.

$m \in \mathbb{Z}^+$, $a, b \in \mathbb{Z}$.
a er kongruent med b modulo m, skrives $a \equiv b \pmod{m}$,
hvis $m \mid a - b$. $\quad \Leftrightarrow \quad m \mid b - a$

$a \equiv b \pmod{m} \;\Leftrightarrow (a \mod m) = (b \mod m)$.

EKS $\quad a = 17, \quad d = 7$

$$17 = 7 \cdot 2 + 3 \qquad 17 \bmod 7 = 3$$

$$-17 = 7 \cdot (-3) + 4 \qquad -17 \bmod 7 = 4$$

EKS

$$31 \equiv 17 \pmod 7 \qquad \text{da } 7 \mid 31 - 17 = 14 = 7 \cdot 2$$

Hvis $a \equiv b \pmod{m}$ og $c \equiv d \pmod{m}$ så er

$$a + c \equiv b + d \pmod{m}$$

$$ac \equiv bd \pmod{m}$$

I udregninger modulo m kan et tal a altså erstattes af (f.eks.) $a \bmod m$.

(Dette gælder ikke tal i en eksponent.)

EKS $m = 3$:

$$4 \cdot 5 + 7 \equiv 1 \cdot 2 + 1 \pmod{3}$$

$$= 3 \equiv 0 \pmod{3}$$

Altfel $3 \mid 4 \cdot 5 + 7$

$$2^5 \not\equiv 2^2 \pmod 3$$