Aufgabenblatt 1 - Aufgabe 3

21. Oktober 2014

(a) Zu beweisen: Es gilt für $n \geq 0$

$$\begin{pmatrix} F_n \\ F_{n+1} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}^n \cdot \begin{pmatrix} F_0 \\ F_1 \end{pmatrix}$$

<u>Induktionsanfang:</u> $n = 0$

$$\begin{aligned} \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}^0 \cdot \begin{pmatrix} F_0 \\ F_1 \end{pmatrix} &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} F_0 \\ F_1 \end{pmatrix} \\ &= \begin{pmatrix} F_0 \\ F_1 \end{pmatrix} \square \end{aligned}$$

<u>Induktionsbehauptung:</u> Es gilt für n:

$$\begin{pmatrix} F_n \\ F_{n+1} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}^n \cdot \begin{pmatrix} F_0 \\ F_1 \end{pmatrix}$$

<u>Induktionsschritt:</u> $n \to n + 1$

$$\begin{aligned} \begin{pmatrix} F_{n+1} \\ F_{n+2} \end{pmatrix} &= \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} F_n \\ F_{n+1} \end{pmatrix} \\ &= \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} \cdot \left[\begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}^n \cdot \begin{pmatrix} F_0 \\ F_1 \end{pmatrix} \right] \\ &= \left[\begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}^n \right] \cdot \begin{pmatrix} F_0 \\ F_1 \end{pmatrix} \\ &= \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}^{n+1} \cdot \begin{pmatrix} F_0 \\ F_1 \end{pmatrix} \end{aligned}$$

(b) Man berechen X^n, indem man erst X^2 berechne, dies mit sich selbst multipliziere ($\widehat{=} X^4$), dies wiederum mit sich selbst multipliziere ($\widehat{=} X^8$)...

Damit benötigt man zB für X^{64} nur 6 Multiplikationen, dies entspricht einem Aufwand von $O(log_2(n))$.

(c) Um F_n mit dem Matrizen-Verfahren zu berechnen benötigt man $(n-1)\cdot$ (8 Multiplikationen und 4 Additionen) um $\begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}^n$ zu berechnen, sowie 4 Multiplikationen und 2 Additionen, um das Ergebnis mit $\begin{pmatrix} F_0 \\ F_1 \end{pmatrix}$ zu verrechnen.

Dies ergibt eine benötigte Zeit von

$$\begin{aligned} & O((n-1)\cdot(8\cdot 64^{1.59}+4\cdot 64)+4\cdot 64^{1.59}+2\cdot 64) \\ =& O((n-\frac{1}{2})\cdot 8\cdot 64^{1.59}+(n-\frac{1}{2})\cdot 4\cdot 64) \\ =& O((8n-4)\cdot(64^{1.59+32})) \end{aligned}$$

Dies ist immer noch linearer Aufwand, und damit asymptotisch echt schneller als $O(n^2)$