Graded homework 09 – Fibonacci matrix Team-Berger-Nussbaum

1) Prove Matrix equality by induction on n

1) Prove matrix equality by induction

$$\begin{bmatrix} F(n-1) & F(n) \\ F(n) & F(n+1) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}^n$$

<u>Verankerung</u>

$$P(1) = \begin{bmatrix} F(1-1) & \boxed{F(1+0)} \\ \boxed{F(1+0)} & F(1+1) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}^1$$

$$= \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 0 & \boxed{1} \\ \boxed{1} & 1 \end{bmatrix} \checkmark$$

<u>Induktionsvoraussetzung</u>

Wir nehmen an, dass $P(n)$ für $n \in \mathbb{N} \setminus \{0\}$ richtig ist:

$$\begin{bmatrix} F(n-1) & F(n) \\ F(n) & F(n+1) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}^n$$

<u>Vererbung</u>

Wir müssen zeigen, dass $P(n+1)$ richtig ist, dass heisst, dass gilt:

$$\begin{bmatrix} F(n) & F(n+1) \\ F(n+1) & F(n+2) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}^{n+1}$$

Zu beginn muss die Induktionsvoraussetzung eingesetzt werden:

$$\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}^{n+1} = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}^n \cdot \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}^1 \overset{*}{=} \begin{bmatrix} F(n-1) & F(n) \\ F(n) & F(n+1) \end{bmatrix} \cdot \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} F(n) & F(n-1)+F(n) \\ F(n+1) & F(n)+F(n+1) \end{bmatrix}$$

$$= \begin{bmatrix} F(n) & F(n+1) \\ F(n+1) & F(n+2) \end{bmatrix} \checkmark$$

* per Induktionsvoraussetzung

Damit ist die Behauptung bewiesen.

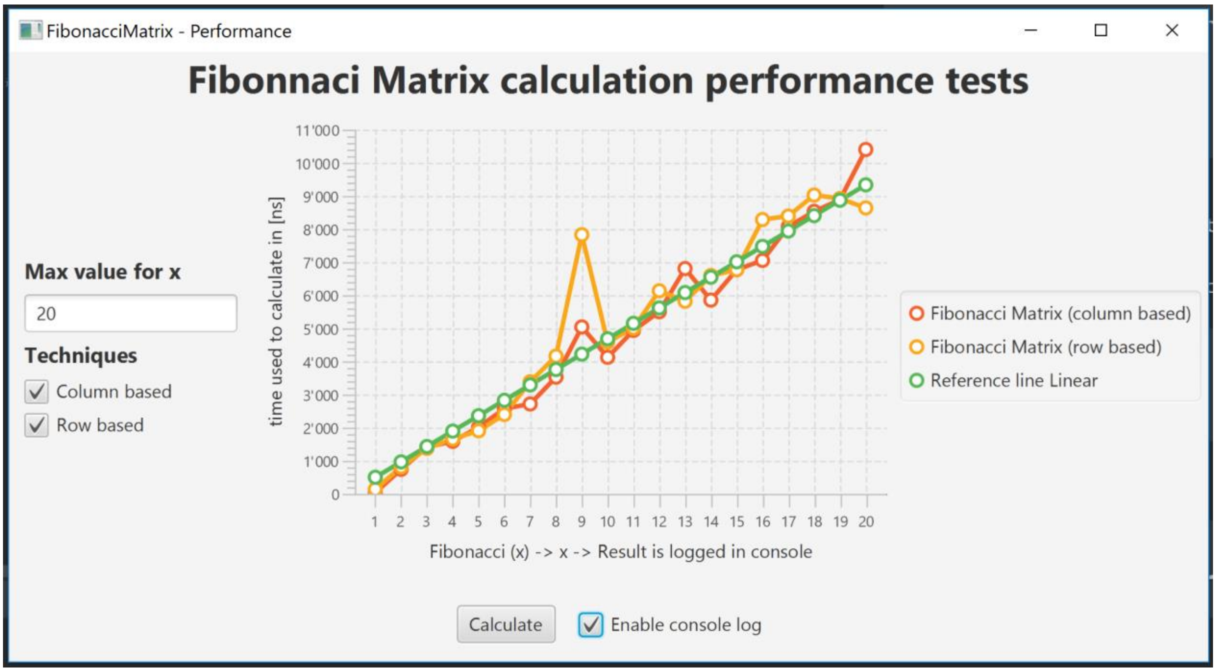
2) Progam matrix-based calculation

See in folder «code» OR
https://github.com/kucki10/BTI7062-Algorithms-and-DataStructures/tree/master/09-homework/Algorithms

3) Print functional graph

Execute the application, or have a look at the following image (Sample)

NOTE: There is only the linear reference line, because we just realized that the complexity is linear.



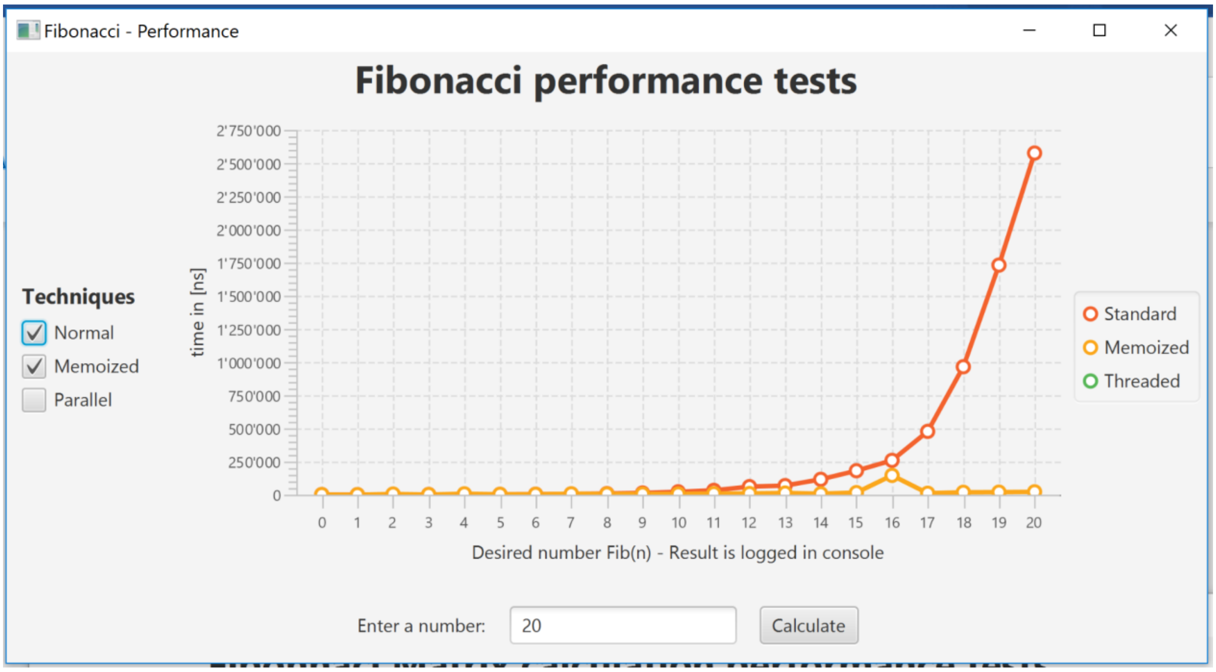
4) Determine the time complexity

The complexity is linear. Explanations are documented here:

https://github.com/kucki10/BTI7062-Algorithms-and-DataStructures/tree/master/09-homework/Algorithms#lessons-learned

5) Compare complexity with Fibonacci algorithms of introductory lecture

Algorithms of introductory lecture



Standard	$\Theta(n^2)$	quadratic
Memoized	$\Theta(n)$	linear (Steigung = 450)
Threaded	-	nicht erkennbar
Fibonacci matrix	$\Theta(n)$	**linear (Steigung = 400)**

The winner is the matrix calculated

6) Swap column and rows in matrix.

Explanatios or findings can be found here:

https://github.com/kucki10/BTI7062-Algorithms-and-DataStructures/tree/master/09-homework/Algorithms#lessons-learned

Details are available in code.