

# Automaten und Komplexität

## Aufgabenzettel 3

Abgabe bis spätestens 12.06.

### Übungsaufgabe 3.1 (Fibonacci-Wörter):

Die Postschen Normalsysteme wurden in der Vorlesung kurz eingeführt. Sie unterscheiden sich von den Semi-Thue Systemen durch ihre Ersetzungsrelation. Die Anwendung einer Regel  $u \rightarrow v$  bewirkt, dass  $u$  von linken Rand des abgeleiteten Wortes gestrichen und  $v$  an den rechten Rand angehängt wird. Es gilt also

$$x \Rightarrow y \quad \text{wenn} \quad x = uw, \quad y = vw \quad \text{und} \quad u \rightarrow v \in P.$$

Insbesondere kann  $u \rightarrow v$  nur auf das Wort  $x$  angewandt werden, wenn  $u$  ein Präfix von  $x$  ist.

Finde ein Postsches Normalsystem, welches die Menge der Fibonacci-Wörter generiert. Zur Erinnerung: Das  $i$ -te Fibonacci-Wort ergibt sich durch  $i$ -malige Anwendung des Homomorphismus'  $h : \begin{matrix} a \rightarrow ab \\ b \rightarrow a \end{matrix}$  auf den Buchstaben  $a$ .

### Übungsaufgabe 3.2 (Kürzeste Pfade in Graphen):

Gegeben ist ein Graph mit  $n$  Knoten und positiven Kantengewichten als Adjazenzmatrix. Also  $a_{ij} = \infty$ : keine Kante, sonst Kante zwischen  $i$  und  $j$  mit Gewicht  $a_{ij}$ .

Schreiben Sie ein Programm für eine PRAM, welches eine Matrix mit dem (als Summe der Gewichte aller benutzten Kanten) kürzesten Pfad zwischen allen Paaren von Knoten zurückgibt. An Zeit steht  $O(\log^2 n)$  und an Prozessoren  $O(n^3)$  zur Verfügung.

Hinweis: Bilden Sie zuerst alle kürzesten Pfade mit nur einer Kante, dann die mit höchstens zwei Kanten, dann die mit vier etc.

### Übungsaufgabe 3.3 (Paralleles Sortieren):

Schreiben Sie ein Programm für eine CREW PRAM, welches  $n$  verschiedene, ganze Zahlen in der Zeit  $O(\log n)$  sortiert und dazu  $O(\frac{n^2}{\log n})$  Prozessoren benötigt. Man gehe wie folgt vor:

1. Erstellung einer Matrix, die die Relation  $<$  für die Zahlen darstellt. Dies gelingt sicher in  $O(1)$  mit  $O(n^2)$  Prozessoren, aber auch in  $O(\log n)$  mit weniger Prozessoren.
2. Bestimmung des  $\pi$  Wertes einer Zahl, d.h der Position in der Reihe der Zahlen, an der sie zu stehen kommen soll.
3. Das Umsortieren auf die Zielposition.