Universität Greifswald
Institute für Mathematik and Informatik
Lecturer: Marc Hellmuth Tutor: Nikolai Nøjgaard

5. Exercise "Datenstrukturen und Effiziente Algorithmen", WS 18/19

Exercise 1: (5 Credits)
Construct for each $k \in \{0, \ldots, m-2\}$ a string of length m such that its suffix tree has $k+m$ edges. Proof your result.

Exercise 2: (4.5+4.5+6=15 Credits)
Use the existence of a linear-time algorithm to construct a (compressed) suffix tree to show that the following problems can be solved in linear-time.

(a) Find *all* occurrences of pattern P in a string S.

(b) Determine whether pattern P occurs in all strings $S_1, \ldots, S_\ell$.

(c) Determine the longest substring of a string S that occurs at least two times in S.

Exercise 3: (2.5+2.5=5 Credits)
Let m be an arbitrary integer and $\mathcal{S}$ be the set of strings $S = s_1 s_2 \ldots s_m$ with $s_i \neq s_m = \$$, $1 \leq i < m$. Provide a string $S \in \mathcal{S}$ such that its implicit suffix

(a) has the fewest number

(b) has the largest number

of edges among all strings in $\mathcal{S}$. Proof your result.
HINT: Exercise 1

Exercise 4: (5 Credits)
Build the implicit suffixtree $\mathcal{T}$ of the string $S\$ = TOCOTOC\$$ using Ukkonens algorithm. Give for each phase $i+1$ and each extension j where Rule 1 is not applied the respective "intermediate" trees.
Draw all suffix-links within your constructed $\mathcal{T}$.

Deadline: Wednesday - November 21, 2018 - 12.15pm