Sequence Analysis 3

Universität Bielefeld, SS 2024
Prof. Dr. Jens Stoye · Dr. Lucas Robidou[1]
https://gi.cebitec.uni-bielefeld.de/teaching/2024summer/sa3

Exercise sheet 5, 16.5.2024 - Discussion 23.05.2024

Exercise 1 (Apply the skew Algorithm)
Apply the *skew* Algorithm, including recursive steps to $S = aaaaa$.

Exercise 2 (Why the recursion works (adapted from Ohlebusch, 2013))
For $i \neq j$ with $i \not\equiv 1 \mod 3$ and $j \not\equiv 1 \mod 3$, prove that $S_i < S_j$ if and only if $\bar{S}_{\tau(i)} < \bar{S}_{\tau(j)}$ where $\bar{S}$ is the concatenation of $[\bar{i}|i \leq 1 \leq n, i \equiv 2 \mod 3]$ and $[\bar{i}|i \leq 1 \leq n, i \equiv 3 \equiv 0 \mod 3]$ with $\bar{i}$ the lexicographic name of the 3-mer at position i and

$$\tau(i) = \begin{cases} \frac{i+1}{3} \text{ if } i \equiv 2 \mod 3 \\ \frac{n+i}{3} \text{ if } i \equiv 0 \mod 3 \end{cases} .$$

Spoilers: `https://gi.cebitec.uni-bielefeld.de/teaching/2024summer/sa3/e5spoilers/`

Exercise 3 (Why 3-mers and not 2-mers?)
You may have been wondering (as I have) why the skew algorithm uses 3-mers and not 2-mers. Try to modify the algorithm, such that it uses 2-mers instead, that is, split the suffix set into two parts based on whether $i \equiv 1 \mod 2$ and then sort the Suffixes with $i \not\equiv 1 \mod 2$ based on the lexicographic name of the 2-mers, etc.
Where does this procedure fail?

[1] If you have questions about this exercise sheet, contact Leonard Bohnenkämper