# Sequence Analysis

# Freie Universität Berlin, Institut für Informatik Prof. Dr. Knut Reinert, Dr. David Weese, Prof. Dr. Peter Clote Sommersemester 2014

1. Exercise sheet, 15. April 2014 Discussion: 22. April 2013 (Ex. 1-3) and 24. April 2013 (Ex. 4-5)

Exercise 1.

Efficient searching with suffix arrays

In the lecture we discussed two strategies how to reduce the number of redundant character comparisons during a binary search. One uses the mlr values, while the other one makes use of lcp values. The mlr trick in practice already brings the running time to  $O(m + \log n)$ .

- Find a pair of pattern and text where the mlr trick still needs time  $O(m \log n)$ .
- For the same text and pattern perform the binary search using the lcp values.

## Exercise 2.

Efficient searching with suffix arrays:

• Prove that using the lcp method the search algorithm does at most  $O(m + \log n)$  character comparisons.

#### Exercise 3.

Suffix array construction:

• What is the worst-case runtime (number of character comparison) when the suffix array is computed with the quicksort algorithm?

#### Exercise 4.

Prove the following assumption stated in the script:

• For the fixed binary search tree used in the search for LP and RP compute the lcp values for its internal nodes using the array height . **The value at an internal node is the minimum of its successors** 

### Exercise 5.

Given a text *T* of length *n*, let *suftab*' be the suffix array of *T* where suffixes are lexicographically ordered according to the first *m* letters for some m < n. Will the Kasai algorithm still compute the correct lcp values of adjacent suffixes in *suftab*'? Justify your answer!