Universität Bielefeld
Technische Fakultät
AG Genominformatik

Dr. Marília D. V. Braga
Prof. Dr. Fábio Martinez
Prof. Dr. Jens Stoye

Advanced Sequence Analysis
Summer 2019

Exercises

Number 1 (12.04.2019), Discussion: 26.04.2019

1. If they exist, find words

   (a) of lengths 2, 4 and 10 that are primitive;

   (b) of lengths 2 and 5 with a period of length 1;

   (c) of lengths 5 and 10 whose only period is of length 4;

   (d) of lengths 6 and 10 whose only periods are of lengths 3 and 5.

2. Compute the

   (i) Hamming distance

   (ii) subword distance

   (iii) edit distance

   (iv) prefix distance

   between the following pairs of words:

   (a) `aabbabaabababa` and `abbaabaabaabba`,

   (b) `abababaaababab` and `abbbbaaabbab`.

3. What is the worst case running time for the Naive String Matching algorithm discussed in class? Give an example where this runnig time is achieved. Present a more efficient algorithm that has optimal running time.

4. Modify the algorithm for computing the subword distance such that it computes the edit distance.

5. Develop a polynomial-time algorithm that finds the longest increasing subsequence in an array X of n numbers, $X[1, \ldots, n]$.

   Hint: Use dynamic programming using an auxiliary array L storing in $L[i]$ the length of the longest increasing subsequence of the prefix $X[1, \ldots, i]$.