58093 String Processing Algorithms (Autumn 2010)
Exercises 5 (9 December)

1. Let $T = \texttt{lallilla\$}$.
   (a) Give the suffix tree of T including suffix links.
   (b) Give the suffix array of T together with the LCP array.

2. Write a pseudocode algorithm for finding all occurrences of a pattern P in a text T using the suffix tree of T.

3. The reverse of a string $S[0..m)$ is the string $S^R = S[m-1]S[m-2]..S[0]$. Describe an algorithm for finding the longest factor S of T such that the reverse S^R is a factor of T too. The method should work in linear time.

4. Give a linear time algorithm for computing the matching statistics of T with respect to S from the generalized suffix *array* of S and T and the associated LCP array (without constructing the suffix tree).

5. Hamming distance is the edit distance with substitution as the only allowed edit operation. Let $ed_H(A, B)$ denote the Hamming distance of two strings A and B of the same length.
   (a) Suppose we have preprocessed the strings A and B so that the longest common extension for any pair of suffixes can be computed in constant time. Show how the Hamming distance $ed_H(A, B)$ can be computed in $\mathcal{O}(ed_H(A, B))$ time.
   (b) Design an $\mathcal{O}(kn)$ worst case time algorithm for approximate string matching with Hamming distance.

6. Prove Lemma 4.9. *Hint:* Generalize Lemma 3.17(b) (Lecture 6) from three strings to many strings.

7. What is the number of distinct factors in the string `abracadabra`?

8. Fill the course feedback form at `https://ilmo.cs.helsinki.fi/kurssit/servlet/Valinta?kieli=en`