

## Tutorial 7 for COMP 526 – Applied Algorithmics, Spring 2021

### Problem 1 (Suffix trees and friends – Part II)

Consider the text  $T = \text{abbabbaa\$}$  again.

Complementing last week's tutorial, construct the

1. suffix array  $L[0..n]$  of  $T$ ,
2. the inverse suffix array  $R[0..n]$ , and
3. the LCP array.
4. Now reconstruct the suffix tree from the above arrays.  
(You can check your construction using the solutions from last week, but try to construct the suffix tree from  $L$  and  $LCP$  first.)
5. Annotate the internal nodes in the suffix tree with their string depth. Explain the connection between string depths and the LCP array.
6. Use the above structures to find the longest repeated substring in  $T$ .

### Problem 2 (Huffman code)

Compress the text  $T = \text{HANNAHBANSBANANASMAN}$  using a Huffman code; give

1. the character frequencies,
2. a step-by-step construction of the Huffman tree,
3. the Huffman code, and
4. the encoded text.
5. Finally, compute the compression ratio of the result  
(ignoring space needed to store the Huffman code).