58093 String Processing Algorithms (Autumn 2010)
Practice problems

1. Each of the following pairs of concepts are somehow connected. Describe the main connecting factors or commonalities as well as the main separating factors or differences.

   (a) Shift–Or algorithm and Myers' bitparallel algorithm.

   (b) LSD radix sort and MSD radix sort.

   (c) Longest common prefix and distinguishing prefix.

   (d) Karp–Rabin algorithm and Karp–Miller–Rosenberg naming technique.

   A few lines for each part is sufficient.

2. Let T be a string and let R be a multiset of symbols. A factor S of T is an occurrence of R if S consists of exactly the symbols of R. For example, if $T = \texttt{abahgcabah}$ and $R = \{\texttt{a}, \texttt{a}, \texttt{b}, \texttt{c}\}$, the only occurrence of R in T is the factor $S = \{caba\}$. Describe an algorithm for finding all occurrences of R in T. The time complexity should be $\mathcal{O}(|T| + |R|)$ on an alphabet of constant size.

3. Construct the Aho–Corasick automaton for the pattern set $\{\texttt{string}, \texttt{ring}, \texttt{trie}, \texttt{log}, \texttt{ecology}\}$. Simulate the scanning of the text `stringology` with the automaton.

4. Define the suffix link in suffix trees and describe briefly its role in a linear time suffix tree construction algorithm.

5. Let $\mathcal{R} = \{S_1, S_2, \ldots, S_k\}$ be a set of strings over a constant size alphabet such that no string in $\mathcal{R}$ is a factor of another string in $\mathcal{R}$. The *shortest distinguishing factor* of S_i is the shortest string that occurs in S_i but not in any other string in $\mathcal{R}$. Describe an algorithm for finding the shortest distinguishing factor for all strings in $\mathcal{R}$. The time complexity should be $\mathcal{O}(||\mathcal{R}||)$, where $||\mathcal{R}||$ is the total length of the strings in $\mathcal{R}$.

6. `ntmaa$iin` is the Burrows–Wheeler transform of which string? (Note that there was an error in the inverse BWT algorithm. The corrected version is now available.)