58093 String Processing Algorithms (Autumn 2011)
Exercises 5 (29–30 November)

1. Let $\mathcal{P} = \{P_1, \ldots, P_{2k}\}$ be a set of patterns, where $P_i = \mathtt{a}^i$ for $i \in [1..k]$, and $|P_j| = 2k$ and P_k is a suffix of P_j for all $j \in [k+1..2k]$.

   (a) Show that the total size of the sets *patterns*$(\cdot)$ in the Aho–Corasick automaton for $\mathcal{P}$ is asymptotically larger than $||\mathcal{P}||$.

   (b) Describe how to represent the sets *patterns*$(\cdot)$ so that
   - the total space complexity is never more than $||\mathcal{P}||$
   - each set *patterns*$(\cdot)$ can be listed in linear time in its size.

2. Show that edit distance is a *metric*, i.e., that it satisfies the metric axioms:
   - $ed(A, B) \geq 0$
   - $ed(A, B) = 0$ if and only if $A = B$
   - $ed(A, B) = ed(B, A)$ (symmetry)
   - $ed(A, C) \leq ed(A, B) + ed(B, C)$ (triangle inequality)

3. Describe a family of string pairs (A_i, B_i), $i \in \mathbb{N}$, such that $|A_i| = |B_i| \geq i$ and there is at least i different optimal edit sequences corresponding to $ed(A_i, B_i)$. Can you find a family, where the number of edit sequences grows much faster than the lengths of the strings?

4. Give a proof for Lemma 4.15 in the lecture notes.

5. Let $P = \mathtt{evete}$ and $T = \mathtt{neeteneeveteen}$.

   (a) Use Ukkonen's cut-off algorithm to find the occurrences of P in T for $k = 1$.

   (b) Simulate the operation of Myers' bitparallel algorithm when it computes column 5 for P and T.

6. A string S is a *subsequence* of a string T if we can construct S by deleting characters from T. Let $lcss(A, B)$ denote the length of the longest common subsequence of the strings A and B. For example, $lcss(\mathtt{berlin}, \mathtt{helsinki}) = 4$ since $\mathtt{elin}$ is a subsequence of both strings.

   (a) Let $ed_{\text{indel}}(A, B)$ be a variant of the edit distance, where insertions and deletions (indels) are the only edit operations allowed (i.e., no substitutions). Show that

   $$ed_{\text{indel}}(A, B) = |A| + |B| - 2 \cdot lcss(A, B)$$

   (b) Give an algorithm for computing $lcss(A, B)$ in time $\mathcal{O}(|A||B|)$.