58093 String Processing Algorithms (Autumn 2014)
Exercises 4 (November 18)

1. The Knuth–Morris–Pratt algorithm differs from the Morris–Pratt algorithm only in the failure function, which can be defined as

   $fail_{\text{KMP}}[i] = k$, where k is the length of the longest proper border of $P[0..i)$ *such that* $P[k] \neq P[i]$, or -1 if there is no such border.

   (a) Compute both failure functions for the pattern `ananassana`.

   (b) Give an example of a text, where some text character is compared three times by the MP algorithm but only once by the KMP algorithm when searching for `ananassana`.

2. Modify Algorithm 2.6 on the lecture notes to compute $fail_{\text{KMP}}$ instead of $fail_{\text{MP}}$.

3. Let us analyze the average case time complexity of the Horspool algorithm, where the average is taken over all possible patterns of length m and all possible texts of length n for the integer alphabet $\Sigma = \{0, 1, \ldots, \sigma - 1\}$ where $\sigma > 1$. This is the same as the expected time complexity when each pattern and text character is chosen independently and randomly from the uniform distribution over Σ.

   (a) Show that the average time spent in the loop on line 7 is $\mathcal{O}(1)$.

   (b) Show that the probability that the shift is shorter than $\min(m, \sigma/2)$ is at most $1/2$.

   (c) Combine the above results to show that the average time complexity is $\mathcal{O}(n/\min(m, \sigma))$.

4. Simulate the execution of the BNDM algorithm for the pattern `anna` and the text `bananamanna`.

5. Show how the following (single) exact string matching algorithms can be modified to solve the *multiple exact string matching problem*:

   (a) Shift-And

   (b) Karp-Rabin

   The solution should be more efficient than the trivial one of searching each pattern separately.