5 Algorithms I (FMS)

One of several ways to perform string matching efficiently is with a finite state automaton (FSA).

(a) Give a brief but clear explanation of the FSA string matching algorithm, its complexity and any associated data structures. [Note: pseudocode of up to 10 lines is allowed, but not required.] [4 marks]

(b) Build the FSA that will find matches of the pattern $P = \text{pepep}$ in an arbitrary string $T$ over the alphabet \{e, o, p\}, explaining what you do and why. [6 marks]

(c) The correctness proof of the FSA string matching algorithm involves the function $\sigma_P(x)$, which is parametric in the pattern $P$ and takes as input a string $x$. Define $\sigma_P(x)$, explaining what it returns. [1 mark]

(d) Let $A, B, C, D$ be character strings; let $|A|$ be the length of string $A$; let $+$ denote integer addition or string concatenation depending on its operands. Let $D$ be the longest suffix of $A$ that is a prefix of $B$.

For each of the following claims: either prove the claim correct, or give a counterexample that proves it is incorrect. You may draw an explanatory picture if it helps clarity.

(i) $\sigma_B(A) = D$ [3 marks]

(ii) $\sigma_B(A + C) = |D| + |C|$ [3 marks]

(iii) $|C| = 1 \implies \sigma_B(A + C) = \sigma_B(A) + 1$ [3 marks]