HW 1 - Computational Models - Spring 2014

Notation: We denote by $\#_\sigma(w)$ the number of times the word $\sigma \in \Sigma^*$ is a substring in the word $w \in \Sigma^*$.

1. For each of the following languages over $\Sigma = \{0, 1\}$, present a drawing representing a DFA that accepts it (correctness proof not needed):

   (a) Σ^*

   (b) $\{11\}||\{0\}^*$

   (c) $\{\epsilon, 0, 10\}$

   (d) $\{w|\ w$ does not contain $'110'\}$

   (e) $\{w|\ w$ contains $'10'$ and doesn't contain $'001'\}$

   (f) $\{xy|\ \#_1(x) \bmod 2 = 1$ and $\#_0(y) \bmod 2 = 1\}$

   (g) $\{w|\ |w| \bmod 3 = 0\}$

   (h) $\{w|\ w$ contains exactly five '0's$\}$

   (i) The complement of $((\{1\} \cup \{01\} \cup \{001\})^*||(\epsilon \cup \{0\} \cup \{00\}))$

2. For the language $\{w|\ w$ contains $'00'\}$ over $\Sigma = \{0, 1\}$, present a drawing representing a DFA that accepts it. Prove correctness.

3. Write a program (Python or Scheme) that includes

   (a) Definition of a data structure representing a DFA. Note: the set of states and the alphabet may each be represented by a natural number.

   (b) A function that given a DFA representation and a word in the alphabet returns a boolean indicating whether the word was accepted by the DFA or not.

   (c) Definition of a data structure representing a NFA.

   (d) A function that given a NFA representation and a word in the alphabet returns a boolean indicating whether the word was accepted by the NFA or not.

4. Present an NFA and convert it to a DFA for the following languages over $\Sigma = \{0, 1\}$:

   (a) $\{w|\ w$ contains $'010'$ or doesn't contain $'101'\}$

   (b) $\{xy|\ \#_0(x) \bmod 3 = 0$ and $\#_0(y) \bmod 2 = 0\}$

5. Present a regular expression for the following languages over $\Sigma = \{0, 1\}$:

   (a) $\{w|\ |w| \bmod 3 = 0\}$

   (b) $\{w|\ w$ contains exactly two '1's$\}$

6. Given that L is a regular language over some alphabet Σ, prove that the following languages are regular:

   (a) $\{xy|\ (x \in L)\ XOR\ (y \in L)\}$

   (b) $\{xy|\ yx \in L\}$