Exercise 2 - Computational Models - Spring 2014

Note1: We denote by $\#_\sigma(w)$ the number of times the word $\sigma \in \Sigma^*$ is a substring in the word $w \in \Sigma^*$.

Note2: You may freely use results from the lectures and recitations.

1. Determine whether the following languages are regular. Prove your answer.
   (a) $L_1 = \{w \mid \exists n \in \mathbb{N}\ s.t.\ |w| = n!\}$ over $\Sigma = \{a\}$.
   (b) $L_2 = \{uvw \mid u, v, w \in \Sigma^* \wedge |uv| = |w| \wedge \#_0(u) = \#_0(v)\}$ over $\Sigma = \{0, 1\}$.
   (c) $L_3 = \{w \mid \#_{010}(w) = \#_{101}(w)\}$ over $\Sigma = \{0, 1\}$.
2. Prove that the following languages are not regular **without using the pumping lemma**.
   (a) $L_1 = \{w \mid \#_a(w) \geq \#_b(w)\}$ over $\Sigma = \{a, b, c\}$.
   (b) $L_2 = \{a^3 b^{n-3} c^n \mid n > 3\}$ over $\Sigma = \{a, b, c\}$.
   (c) $L_3 = \{a^n b a^n \mid n \geq 0\}$ over $\Sigma = \{a, b, c\}$.
3. Determine whether the following languages are context free. Prove your answer.
   (a) $L_1 = \{a^{2n} b^{3n} \mid n \in \mathbb{N}\}$ over $\Sigma = \{a, b\}$.
   (b) $L_2 = \{0^p \mid p \in Primes\}$.
   (c) $L_3 = \{wcv \mid w\ is\ a\ substring\ of\ v\}$ over $\Sigma = \{a, b, c\}$.
   (d) $L_4 = \{w \mid w \neq w^R\}$ over $\Sigma = \{0, 1\}$.
4. For a binary word w we define $\overline{w}$ as the word with every bit flipped. For example $\overline{010}$ is 101. Let $L = \{w\overline{w} \mid w \in \{0, 1\}^*\}$.
   (a) Use the pumping lemma to prove that L is not CFG.
   (b) Find a context free grammar for $\overline{L}$. What does it imply about context free languages?

5. Let $DropMiddle(L) = \{xy \in \Sigma^* \mid |x| = |y| \wedge \exists a \in \Sigma, xay \in L\}$. Prove that context free languages are not closed under DropMiddle.

6. This question deals with algorithmic problems.

   (a) Describe an algorithm that given a DFA A, decides if $L(A)$ is infinite. (possible hint: use the pumping lemma).

   (b) Describe an algorithm that given a DFA A, decides if $|L(A)| = 9,122,009$.

   (c) Describe an algorithm that given two DFAs A_1 and A_2, decides if $L(A_1) = L(A_2)$.

   (d) Describe an algorithm that given a CFG G, decides if $L(G)$ is infinite.