Tutorial 5

Exercise 1: Consider the following pushdown automaton $\mathcal{M} = (Q, \Sigma, \Gamma, \delta, q_1, C)$ where $Q = \{q_1, q_2\}$, $\Sigma = \{a, b, c\}$, $\Gamma = \{A, B, C\}$, and where the transition function δ is given by the following set of rules:

$$\begin{array}{llll}
q_1C \xrightarrow{a} q_1A & q_1A \xrightarrow{a} q_1AA & q_1B \xrightarrow{a} q_1AB & q_2A \xrightarrow{a} q_2 \\
q_1C \xrightarrow{b} q_1B & q_1A \xrightarrow{b} q_1BA & q_1B \xrightarrow{b} q_1BB & q_2B \xrightarrow{b} q_2 \\
q_1C \xrightarrow{c} q_2 & q_1A \xrightarrow{c} q_2A & q_1B \xrightarrow{c} q_2B &
\end{array}$$

The automaton $\mathcal{M}$ accepts by an empty stack.

Write down the sequence of all configurations through which the automaton $\mathcal{M}$ goes during its computation on word $abaacaaba$.

Exercise 2: For each of the following languages construct a pushdown automaton accepting the given language.

The constructed automata can be nondeterministic. For those languages where it is possible, try to construct a deterministic pushdown automaton.

Remark: It can be reasonable to start with an informal description of behaviour of your proposed automaton for the given language. This description should be detailed enough to make clear how the automaton works. Then for at least some of these languages finish the detailed constuction of the automaton where you write down the formal description of the automaton, i.e., all its states, transitions, etc.

a) $\{ww^R \mid w \in \{a, b\}^*\}$

b) $\{a^n b^m \mid n > m\}$

c) $\{a^n b^i cbc^j \mid n = i + j\}$

Remark: The alphabet is $\{a, b, c\}$.

d) $\{w \in \{a, b\}^* \mid |w|_a > |w|_b\}$

e) The complement of the language $\{a^n b^n \mid n \geq 0\}$.

f) $\{wcx \mid w, x \in \{a, b\}^*$ and w^R is a subword of word $x\}$

Remark: The alphabet $\{a, b, c\}$.

g) $\{w_1cw_2c \cdots cw_k \mid k \geq 1$, for each w_i it holds that $w_i \in \{a, b\}^*$, and there are some i and j such that $w_i = w_j^R\}$

Remark: The alphabet is $\{a, b, c\}$. Note also that for the given i and j, it can be true that $i = j$.

Exercise 3: Construct a pushdown automaton $\mathcal{M}$ accepting the language generated by the following context-free grammar $\mathcal{G}$ (i.e., such that $\mathcal{L}(\mathcal{M}) = \mathcal{L}(\mathcal{G})$):

$$\begin{array}{rcl} S & \longrightarrow & \varepsilon \mid AS \\ A & \longrightarrow & aAb \mid B \\ B & \longrightarrow & \varepsilon \mid bB \end{array}$$

Write down a derivation of word $aabbabbb$ in the grammar $\mathcal{G}$ and some accepting computation of the automaton $\mathcal{M}$ over this word.

Can you see some correspondence between this derivation in the grammar $\mathcal{G}$ and the computation of the automaton $\mathcal{M}$?

Exercise 4: Consider the following construction: For the given pushdown automaton $\mathcal{M} = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$, which accepts using accepting states, we construct a pushdown automaton $\mathcal{M}' = (Q, \Sigma, \Gamma, \delta', q_0, Z_0)$ accepting by an empty stack, where δ' constains the same transitions as function δ, to which we add additional transitions of the form

$$qX \xrightarrow{\varepsilon} q$$

for each $q \in F$ and each $X \in \Gamma$.

a) Show that the constuction described above does not in general lead to a construction of an equivalent automaton, i.e., give an example of a concrete pushdown automaton $\mathcal{M}$ such that when we apply this construction to it, we obtain an automaton $\mathcal{M}'$ such that $\mathcal{L}(\mathcal{M}) \neq \mathcal{L}(\mathcal{M}')$.

b) Propose how to modify this construction in such a way that it ensures that the resulting automaton will be always equivalent to the original automaton, i.e., that for an arbitrary pushdown automaton $\mathcal{M}$ accepting by an accepting state this modified construction always produces a pushdown automaton $\mathcal{M}'$ accepting by an empty stack such that $\mathcal{L}(\mathcal{M}) = \mathcal{L}(\mathcal{M}')$.

Exercise 5: Determine for each of the following languages if the given language is (i) regular, (ii) context-free. Justify your answers at least informally.

a) $\{ w \in \{a, b\}^* \mid w$ does not end with suffix baa and $|w|_a \bmod 3 = 2 \}$

b) $\{ a^j \mid j$ is a power of number $2 \}$

c) $\{ w \in \{0, 1\}^* \mid w$ is a power of number 2 written in binary$\}$

d) the language described by regular expression $a^*b^*c^*$

e) $\{ w \in \{a, +\}^* \mid w$ is generated by grammar $S \longrightarrow S{+}S \mid a \}$

f) $\{ w \in \{a, +, (,)\}^* \mid w$ is generated by grammar $S \longrightarrow S{+}S \mid (S) \mid a \}$

g) $\{ a^m b^n \mid (m \bmod 3) > (n \bmod 3) \}$

h) $\{ a^m b^n \mid n \neq m \}$

i) $\{\, a^m b^n \mid m, n \geq 0,\;\; 5m + 3n = 24 \,\}$

j) $\{\, a^m b^n \mid m, n \geq 0,\;\; 5m - 3n = 24 \,\}$

k) $\{\, w \in \{a, b, c\}^* \mid |w|_a = |w|_b = |w|_c \,\}$

l) $\{\, w \in \{a, b, c\}^* \mid |w|_a > |w|_b \text{ and } |w|_b > |w|_c \,\}$

m) $\{\, w \in \{a, b, c\}^* \mid |w|_a > |w|_b \text{ or } |w|_b > |w|_c \,\}$

n) $\{\, a^n b^m c^k d^\ell \mid 2n = 3k \text{ or } 5m = 7\ell \,\}$

o) $\{\, a^n b^m c^k d^\ell \mid 2n = 3k \text{ and } 5m = 7\ell \,\}$

p) $\{\, a^n b^m c^k d^\ell \mid 2n = 3m \text{ and } 5k = 7\ell \,\}$

q) $\{\, a^n b^m c^k d^\ell \mid 2n = 3\ell \text{ and } 5k = 7m \,\}$

r) $\{\, ww \mid w \in \{a, b\}^* \,\}$

s) $\{\, ww^R \mid w \in \{a, b\}^* \,\}$

t) $\{\, ww \mid w \in \{a\}^* \,\}$