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 $abaacaaba$.

Solution:

$$\begin{array}{l}
(q_1,\ abaacaaba,\ C) \\
\quad\longrightarrow\ (q_1,\ baacaaba,\ A) \\
\quad\longrightarrow\ (q_1,\ aacaaba,\ BA) \\
\quad\longrightarrow\ (q_1,\ acaaba,\ ABA) \\
\quad\longrightarrow\ (q_1,\ caaba,\ AABA) \\
\quad\longrightarrow\ (q_2,\ aaba,\ AABA) \\
\quad\longrightarrow\ (q_2,\ aba,\ ABA) \\
\quad\longrightarrow\ (q_2,\ ba,\ BA) \\
\quad\longrightarrow\ (q_2,\ a,\ A) \\
\quad\longrightarrow\ (q_2,\ \varepsilon,\ \varepsilon)
\end{array}
\qquad \text{or} \qquad
\begin{array}{rl}
q_1C & \xrightarrow{a}\ q_1A \\
 & \xrightarrow{b}\ q_1BA \\
 & \xrightarrow{a}\ q_1ABA \\
 & \xrightarrow{a}\ q_1AABA \\
 & \xrightarrow{c}\ q_2AABA \\
 & \xrightarrow{a}\ q_2ABA \\
 & \xrightarrow{a}\ q_2BA \\
 & \xrightarrow{b}\ q_2A \\
 & \xrightarrow{a}\ q_2
\end{array}$$

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

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

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

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

Solution: Pushdown automaton $\mathcal{M} = (Q, \Sigma, \Gamma, \delta, q_0, X)$ accepting by an empty stack, where $Q = \{q_0, q_1, q_2, q_3\}$, $\Sigma = \{a, b\}$, $\Gamma = \{X, A\}$, and where the transition function δ is given by the following set of rules:

$$\begin{array}{lll}
q_0X \xrightarrow{a} q_0AX & q_0X \xrightarrow{b} q_2X & \\
q_0A \xrightarrow{a} q_0AA & q_0A \xrightarrow{b} q_1 & q_0A \xrightarrow{\varepsilon} q_3A \\
q_1X \xrightarrow{a} q_2X & q_1X \xrightarrow{b} q_2X & \\
q_1A \xrightarrow{a} q_2A & q_1A \xrightarrow{b} q_1 & q_1A \xrightarrow{\varepsilon} q_3A \\
q_2X \xrightarrow{a} q_2X & q_2X \xrightarrow{b} q_2X & q_2X \xrightarrow{\varepsilon} q_3X \\
q_2A \xrightarrow{a} q_2A & q_2A \xrightarrow{b} q_2A & q_2A \xrightarrow{\varepsilon} q_3A \\
q_3A \xrightarrow{\varepsilon} q_3 & q_3X \xrightarrow{\varepsilon} q_3 &
\end{array}$$

Remark: The same automaton will work also for acceptance by an accepting state if we choose $F = \{q_3\}$.

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 $i = j$.

Solution: Pushdown automaton $\mathcal{M} = (Q, \Sigma, \Gamma, \delta, q_0, X)$ accepting by an empty stack where $Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6\}$, $\Sigma = \{a, b, c\}$, $\Gamma = \{X, A, B\}$, and where the transition function δ is given by the following set of rules:

$$\begin{array}{llll}
q_0X \xrightarrow{a} q_1X & q_0X \xrightarrow{b} q_1X & q_0X \xrightarrow{c} q_0X & q_0X \xrightarrow{\varepsilon} q_2X \\
q_1X \xrightarrow{a} q_1X & q_1X \xrightarrow{b} q_1X & q_1X \xrightarrow{c} q_0X & \\
q_2X \xrightarrow{a} q_2AZ & q_2X \xrightarrow{b} q_2BZ & q_2X \xrightarrow{c} q_3X & \\
q_2X \xrightarrow{a} q_5X & q_2X \xrightarrow{b} q_5X & & q_2X \xrightarrow{\varepsilon} q_5X \\
q_2A \xrightarrow{a} q_2AA & q_2A \xrightarrow{b} q_2BA & q_2A \xrightarrow{c} q_3A & \\
q_2A \xrightarrow{a} q_5A & q_2A \xrightarrow{b} q_5A & & q_2A \xrightarrow{\varepsilon} q_5A \\
q_2B \xrightarrow{a} q_2AB & q_2B \xrightarrow{b} q_2BB & q_2B \xrightarrow{c} q_3B & \\
q_2B \xrightarrow{a} q_5B & q_2B \xrightarrow{b} q_5B & & q_2B \xrightarrow{\varepsilon} q_5B \\
q_3X \xrightarrow{a} q_4X & q_3X \xrightarrow{b} q_4X & q_3X \xrightarrow{c} q_3X & q_3X \xrightarrow{\varepsilon} q_5X \\
q_3A \xrightarrow{a} q_4A & q_3A \xrightarrow{b} q_4A & q_3A \xrightarrow{c} q_3A & q_3A \xrightarrow{\varepsilon} q_5A \\
q_3B \xrightarrow{a} q_4B & q_3B \xrightarrow{b} q_4B & q_3B \xrightarrow{c} q_3B & q_3B \xrightarrow{\varepsilon} q_5B \\
q_4X \xrightarrow{a} q_4X & q_4X \xrightarrow{b} q_4X & q_4X \xrightarrow{c} q_3X & \\
q_4A \xrightarrow{a} q_4A & q_4A \xrightarrow{b} q_4A & q_4A \xrightarrow{c} q_3A & \\
q_4B \xrightarrow{a} q_4B & q_4B \xrightarrow{b} q_4B & q_4B \xrightarrow{c} q_3B & \\
q_5A \xrightarrow{a} q_5 & q_5B \xrightarrow{b} q_5 & q_5X \xrightarrow{c} q_6X & q_5X \xrightarrow{\varepsilon} q_6 \\
q_6X \xrightarrow{a} q_6X & q_6X \xrightarrow{b} q_6X & q_6X \xrightarrow{c} q_6X & q_6X \xrightarrow{\varepsilon} q_6
\end{array}$$

Remark: The same automaton will work also by acceptance with an accepting state if we choose $F = \{q_6\}$.

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 $\mathcal{M}$?

Solution: Pushdown automaton $\mathcal{M} = (q_0, \Sigma, \Gamma, \delta, q_0, S)$ where $\Sigma = \{a, b\}$, $\Gamma = \{S, A, B, a, b\}$, and where the transition function δ is given by the following set of rules:

$$\begin{array}{ll}
q_0S \xrightarrow{\varepsilon} q_0 & q_0a \xrightarrow{a} q_0 \\
q_0S \xrightarrow{\varepsilon} q_0AS & q_0b \xrightarrow{b} q_0 \\
q_0A \xrightarrow{\varepsilon} q_0aAb & \\
q_0A \xrightarrow{\varepsilon} q_0B & \\
q_0B \xrightarrow{\varepsilon} q_0 & \\
q_0B \xrightarrow{\varepsilon} q_0bB &
\end{array}$$

$$
\begin{aligned}
S &\Rightarrow AS \\
&\Rightarrow aAbS \\
&\Rightarrow aaAbbS \\
&\Rightarrow aaBbbS \\
&\Rightarrow aabbS \\
&\Rightarrow aabbAS \\
&\Rightarrow aabbaAbS \\
&\Rightarrow aabbaBbS \\
&\Rightarrow aabbabBbS \\
&\Rightarrow aabbabbBbS \\
&\Rightarrow aabbabbbS \\
&\Rightarrow aabbabbb
\end{aligned}
$$

$$
\begin{aligned}
q_0S &\xrightarrow{\varepsilon} q_0AS \\
&\xrightarrow{\varepsilon} q_0aAbS \\
&\xrightarrow{a} q_0AbS \\
&\xrightarrow{\varepsilon} q_0aAbbS \\
&\xrightarrow{a} q_0AbbS \\
&\xrightarrow{\varepsilon} q_0BbbS \\
&\xrightarrow{\varepsilon} q_0bbS \\
&\xrightarrow{b} q_0bS \\
&\xrightarrow{b} q_0S \\
&\xrightarrow{\varepsilon} q_0AS \\
&\xrightarrow{\varepsilon} q_0aAbS \\
&\xrightarrow{a} q_0AbS \\
&\xrightarrow{\varepsilon} q_0BbS \\
&\xrightarrow{\varepsilon} q_0bBbS \\
&\xrightarrow{b} q_0BbS \\
&\xrightarrow{\varepsilon} q_0bBbS \\
&\xrightarrow{b} q_0BbS \\
&\xrightarrow{\varepsilon} q_0bS \\
&\xrightarrow{b} q_0S \\
&\xrightarrow{\varepsilon} q_0
\end{aligned}
$$

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 $\mathcal{L}(\mathcal{M}) \neq \mathcal{L}(\mathcal{M}')$.

Solution: For example pushdown automaton $\mathcal{M} = (Q, \Sigma, \Gamma, \delta, q_0, X, F)$, accepting by accepting states, where $Q = \{q_0, q_1, q_2\}$, $\Sigma = \{a, b, c\}$, $\Gamma = \{X, A, B\}$, $F = \{q_2\}$, and where the transition function δ is given by the following set of rules:

$$
\begin{array}{llll}
q_0X \xrightarrow{a} q_1AX & q_1A \xrightarrow{a} q_1AB & q_1B \xrightarrow{b} q_1 & q_2B \xrightarrow{b} q_2B \\
 & q_1A \xrightarrow{b} q_1 & q_1X \xrightarrow{\varepsilon} q_2 & q_2X \xrightarrow{b} q_2X \\
 & q_1A \xrightarrow{c} q_2A & &
\end{array}
$$

This automaton accepts the language

$$L = \{a^n b^n \mid n \geq 1\} \cup \{a^n c \mid n \geq 1\}.$$

After perfoming the transformation described above, the automaton $\mathcal{M}'$ will accept all words from the language L but also all words of the form $a^n c b^m$ where $n \geq 1$ and $m \geq 1$.

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 $\mathcal{L}(\mathcal{M}) = \mathcal{L}(\mathcal{M}')$.

Solution: For a pushdown automaton $\mathcal{M} = (Q, \Sigma, \Gamma, \delta, q_0, X_0, F)$ that accepts by accepting states, we can construct an automaton $\mathcal{M}' = (Q', \Sigma, \Gamma', \delta', q_0', X_0')$accepting by an empty stack where:

- $Q' = Q \cup \{q_0', q_f\}$ where q_0' and q_f are new states that do not occur in Q,
- $\Gamma' = \Gamma \cup \{X_0'\}$ where X_0' is a new pushdown symbol that does not occur in Γ,
- δ' contains all rules from δ and also the rules
  - $q_0' X_0' \xrightarrow{\varepsilon} q_0 X_0 X_0'$,
  - $qX \xrightarrow{\varepsilon} q_f$ for each $q \in F$ and $X \in \Gamma'$,
  - $q_f X \xrightarrow{\varepsilon} q_f$ for each $X \in \Gamma'$.

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 $|w|_a \bmod 3 = 2 \}$

Solution: Regular, context-free.

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

Solution: Non-regular, non-context-free.

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

Solution: Regular, context-free.

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

Solution: Regular, context-free.

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

Solution: Regular, context-free.

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

Solution: Non-regular, context-free.

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

Solution: Regular, context-free.

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

Solution: Non-regular, context-free.

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

Solution: Regular, context-free.

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

Solution: Non-regular, context-free.

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

Solution: Non-regular, non-context-free.

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

Solution: Non-regular, non-context-free.

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

Solution: Non-regular, context-free.

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

Solution: Non-regular, context-free.

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

Solution: Non-regular, non-context-free.

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

Solution: Non-regular, context-free.

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

Solution: Non-regular, context-free.

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

Solution: Non-regular, non-context-free.

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

Solution: Non-regular, context-free.

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

Solution: Regular, context-free.