Solutions for the Midterm Exam

1. Give a regular expression that represents the set of all non-empty strings over \( \Sigma = \{a, b, c\} \) in which all the \( a \)'s precede the \( b \)'s, which in turn precede the \( c \)'s.

   **Solution:**
   
   \[ a^+b^*c^* \cup a^*b^+c^* \cup a^*b^*c^+ \]

   (20 points)

2. Consider the following grammar \( G \):

   \[
   \begin{align*}
   S & \rightarrow XY \\
   X & \rightarrow aX \mid bX \mid a \\
   Y & \rightarrow Ya \mid Yb \mid b
   \end{align*}
   \]

   (a) Give a leftmost derivation of \( abaabb \).
   (b) Build the derivation tree for the derivation in part (a).
   (c) What is \( L(G) \)? Give a regular expression!

   **Solution:**

   (a) The following is a leftmost derivation of \( abaabb \):

   \[
   \begin{align*}
   S & \Rightarrow XY \\
   & \Rightarrow aY \\
   & \Rightarrow aYb \\
   & \Rightarrow aYbb \\
   & \Rightarrow aYabb \\
   & \Rightarrow abaabb
   \end{align*}
   \]
(b)

(c)

\[
L(G) = (a \cup b)^* ab(a \cup b)^*
\]

(20 points)

3. Construct two regular grammars, one ambiguous and one unambiguous, that generate the language determined in the previous problem 2(c).

Solution:

Unambiguous regular grammar:

\[
\begin{align*}
S & \rightarrow bS \mid aA \\
A & \rightarrow aA \mid bB \\
B & \rightarrow aB \mid bB \mid \lambda
\end{align*}
\]

Ambiguous regular grammar:

\[
\begin{align*}
S & \rightarrow bS \mid aA \\
A & \rightarrow aA \mid bB \mid bC \\
B & \rightarrow aB \mid bB \mid \lambda \\
C & \rightarrow aC \mid bC \mid \lambda
\end{align*}
\]

It is ambiguous because there are two different leftmost derivations for the string \(ab\):

\[
\begin{align*}
S & \Rightarrow aA \\
& \Rightarrow abB \\
& \Rightarrow ab
\end{align*}
\]
and

\[ S \Rightarrow aA \]
\[ \Rightarrow abC \]
\[ \Rightarrow ab \]

(20 points)

4. Design a DFA that accepts the language determined in problem 2(c).

(20 points)

**Solution:**

The state diagram of a DFA is

![DFA Diagram](image)

5. Design the state diagram of a DFA equivalent to the following NFA-\( \lambda \) by using the subset construction method. What is the language accepted by these machines? (20 points)

![NFA-\( \lambda \) Diagram](image)
Solution:

Here the states correspond to the following subsets: \( A = \{q_0, q_1, q_2\} \), 
\( B = \{q_1, q_2\} \), \( C = \{q_2\} \) and \( D = \emptyset \). The language is \( a^*b^*c^* \).