QUESTION 1. [10 pts] Design a DFA to accept the following language:

\[ L = \{ x \mid x \in \{0,1\}^*, \text{the number of 0's in } x \text{ is divisible by 4 and the number of 1's in } x \text{ is divisible by 2} \} \]

Answer:

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>*</td>
<td>( q_0,0 )</td>
<td>( q_{1,0} )</td>
</tr>
<tr>
<td>( q_{0,1} )</td>
<td>( q_{1,1} )</td>
<td>( q_{0,0} )</td>
</tr>
<tr>
<td>( q_{1,0} )</td>
<td>( q_{2,0} )</td>
<td>( q_{1,1} )</td>
</tr>
<tr>
<td>( q_{1,1} )</td>
<td>( q_{2,1} )</td>
<td>( q_{1,0} )</td>
</tr>
<tr>
<td>( q_{2,0} )</td>
<td>( q_{3,0} )</td>
<td>( q_{2,1} )</td>
</tr>
<tr>
<td>( q_{2,1} )</td>
<td>( q_{3,1} )</td>
<td>( q_{2,0} )</td>
</tr>
<tr>
<td>( q_{3,0} )</td>
<td>( q_{0,0} )</td>
<td>( q_{3,1} )</td>
</tr>
<tr>
<td>( q_{3,1} )</td>
<td>( q_{0,1} )</td>
<td>( q_{3,0} )</td>
</tr>
</tbody>
</table>

Give partial credits for DFAs that handles the number of 0’s or the number of 1’s correctly.
QUESTION 2. [10 pts] Design an $\epsilon$-NFA to accept the following language:

$$L = \{ x \mid x \in \{0, 1\}^*, \text{ x begins or ends with 101} \}$$

Answer:

Note that the answer is not unique.
QUESTION 3. [10 pts] Convert the following NFA to a DFA:

\[
\begin{array}{c|cc}
 & 0 & 1 \\
\hline
\rightarrow q_0 & \{q_1\} & \{q_0, q_1\} \\
q_1 & \{q_1, q_2\} & \{q_2\} \\
* q_2 & \{q_1\} & \emptyset \\
\end{array}
\]

Answer:

\[
\begin{array}{c|cc}
 & 0 & 1 \\
\hline
\rightarrow \{q_0\} & \{q_1\} & \{q_0, q_1\} \\
\{q_1\} & \{q_1, q_2\} & \{q_2\} \\
* \{q_2\} & \{q_1\} & \emptyset \\
\{q_0, q_1\} & \{q_1, q_2\} & \{q_0, q_1, q_2\} \\
* \{q_1, q_2\} & \{q_1, q_2\} & \{q_2\} \\
* \{q_0, q_1, q_2\} & \{q_1, q_2\} & \{q_0, q_1, q_2\} \\
\emptyset & \emptyset & \emptyset \\
\end{array}
\]

It’s okay to include inaccessible states (i.e., \(\{q_0, q_2\}\)). Give partial credits for correct steps.
QUESTION 4. [10 pts] Give a regular expression for the following language:

\[ L = \{ x \mid x \in \{0,1\}^*, \ x \text{ has at most two } 1\text{'s between consecutive } 0\text{'s} \} \]

For example, the language constains strings 111, 10111, 11011010010111, but not strings like
10111110 or 01110100.

Answer:

\[ 1^* + 1^*0((\epsilon + 1 + 11)0)^*1^* \]

The regex is not unique. Give partial credits for regex’s containing some correct components.
QUESTION 5. [10 pts] Convert the following DFA to a regular expression by eliminating its states in the order $q_2, q_1$:

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>$q_0$</td>
<td>$q_2$</td>
<td>$q_1$</td>
</tr>
<tr>
<td>$q_1$</td>
<td>$q_2$</td>
<td>$q_0$</td>
</tr>
<tr>
<td>$q_2$</td>
<td>$q_1$</td>
<td>$q_2$</td>
</tr>
</tbody>
</table>

Answer:

After eliminating state $q_2$, we have a new transition from $q_1$ to $q_1$ with label 01*0 (2 pts), and the arc label from $q_0$ to $q_1$ becomes $1 + 01^*$ (2 pts).

After eliminating state $q_1$, the arc label from $q_0$ to $q_0$ becomes (3 pts)

$$(1 + 01^*) (01^*)^* 1$$

which leaves the final answer as (3 pts)

$$((1 + 01^*) (01^*)^*)^*$$

Give partial credits for correct steps. Deduct at least 3 pts for eliminating states in a different order.
QUESTION 6. [10 pts] Following the procedure given in class, convert the following regular expression to an $\epsilon$-NFA:

$$R = 0^*(10^* + 11)0^*$$

Answer:

<table>
<thead>
<tr>
<th></th>
<th>$\epsilon$</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\rightarrow q_0$</td>
<td>${q_1}$</td>
<td>$\emptyset$</td>
<td>$\emptyset$</td>
</tr>
<tr>
<td>$q_1$</td>
<td>${q_2}$</td>
<td>${q_1}$</td>
<td>$\emptyset$</td>
</tr>
<tr>
<td>$q_2$</td>
<td>$\emptyset$</td>
<td>$\emptyset$</td>
<td>${q_3, q_4}$</td>
</tr>
<tr>
<td>$q_3$</td>
<td>${q_6}$</td>
<td>${q_3}$</td>
<td>$\emptyset$</td>
</tr>
<tr>
<td>$q_4$</td>
<td>$\emptyset$</td>
<td>$\emptyset$</td>
<td>${q_5}$</td>
</tr>
<tr>
<td>$q_5$</td>
<td>${q_6}$</td>
<td>$\emptyset$</td>
<td>$\emptyset$</td>
</tr>
<tr>
<td>$q_6$</td>
<td>${q_7}$</td>
<td>${q_6}$</td>
<td>$\emptyset$</td>
</tr>
<tr>
<td>$*q_7$</td>
<td>$\emptyset$</td>
<td>$\emptyset$</td>
<td>$\emptyset$</td>
</tr>
</tbody>
</table>

Note that the $\epsilon$-NFA in the answer has been simplified a little bit. There could be a few more states with trivial $\epsilon$-transitions, or fewer states with further simplification.
QUESTION 7. [15 pts] Use the test technique learned in class to prove or disprove the following general identities concerning regular expressions.

1. \((EF)^* = E^*F^*\)
2. \((E + \epsilon)^* = E^*\)

Answer:

1. False (1 pt). Replace \(E\) by symbol \(a\) and \(F\) by symbol \(b\). Then \(LHS = (ab)^*\) and \(RHS = a^*b^*\) (2 pts). They are clearly not equal, because the string \(abab\) is in \(L((ab)^*)\) but not in \(L(a^*b^*)\). (3 pts)

2. True (1 pt). Replace \(E\) by symbol \(a\). Then \(LHS = (a + \epsilon)^*\) and \(RHS = a^*\) (2 pts). Since \(L(a^*)\) is the universe over the alphabet \(\{a\}\), clearly \(L((a + \epsilon)^*) \subseteq L(a^*)\) (3 pts). On the other hand, since \(L(a) \subseteq L(a + \epsilon)\), \(L(a^*) \subseteq L((a + \epsilon)^*)\) (3 pts). Hence, \((a + \epsilon)^* = a^*\).

Give partial credits for major correct ideas/steps.
QUESTION 8. [10 pts] Prove that the following language is not regular using the Pumping Lemma:

\[ L = \{0^i1^j2^{2i} \mid i, j \geq 1\} \]

Answer:

Let \( n \) be the constant in the Pumping Lemma. Consider \( w = 0^n12^{2n} \in L \). Let \( w = xyz \) be any partition satisfying (i) \(|y| > 0\) and \(|xy| \leq n\). Clearly, \( x = 0^s \) and \( y = 0^t \) for some \( t > 0 \). Then \( xyyz = 0^{n+t}12^{2n} \notin L \), and thus a contradiction.

Give partial credits for correct/reasonable steps.