Problem 1.21 (2 pts each)

a) \((a \cup ba^*b)^*ba^*\) or \(a^*b(a \cup ba^*b)^*\)

b) \(((a \cup b)(a \cup bb)^*ab)^*(((a \cup b)(a \cup bb)^*(b \cup ab) \cup E)\) or \(((a \cup b)a^*bba^*a)^*(((a \cup b)ba^*b \cup E)\) or equivalent alternative

Problem 1.28 (2 pts each)

a)

\[\begin{array}{cccccccccccc}
\rightarrow & 0 & \rightarrow & 0 & \rightarrow & \Theta & \rightarrow & 0 & \rightarrow & 0 & \rightarrow & 0 & \rightarrow & 0 & \rightarrow & 0 & \rightarrow & 0 & \rightarrow & 0 & \rightarrow & 0 & \rightarrow & 0 & \rightarrow & 0 & \rightarrow & 0 & \rightarrow & \Theta \\
\end{array}\]

b)

\[\begin{array}{cccccccccccc}
\rightarrow & 0 & \rightarrow & 0 & \rightarrow & \Theta & \rightarrow & 0 & \rightarrow & 0 & \rightarrow & 0 & \rightarrow & 0 & \rightarrow & 0 & \rightarrow & 0 & \rightarrow & 0 & \rightarrow & 0 & \rightarrow & 0 & \rightarrow & 0 & \rightarrow & 0 & \rightarrow & 0 & \rightarrow & 0 & \rightarrow & \Theta \\
\end{array}\]
Problem 1.42 (2 pts)

Let \( D_A = (Q_A, \Sigma, \delta_A, F_A) \) and \( D_B = (Q_B, \Sigma, \delta_B, F_B) \) be two DFAs that recognize languages A and B.

We can then construct an NFA \( N = (Q, \Sigma, \delta, q, F) \) that recognizes the shuffle of A and B.

Define the NFA \( N \) as follows:

a) \( Q = (Q_A \times Q_B) \), where \( Q_A \times Q_B \) represents all possible current states of \( D_A \) and \( D_B \).

b) \( q = (q_A, q_B) \)

c) \( F = (F_A \times F_B) \), which represents that \( N \) accepts a string if both \( D_A \) and \( D_B \) accept it.

d) \( \delta \) is defined as: \( \delta((x, y), a) = \{(\delta_A(x, a), b), (x, \delta_B(y, a))\} \).

Problem 1.46 (2 pts each)

Proof of part a.

We assume that \( A \) is a regular language, so there exists a pumping length \( p \). We choose the string \( s \) to be \( 0^p10^p \), which is in \( A \) and the length of the string is \( 2p + 1 \geq p \). Thus, \( s \) can be divided into 3 pieces \( xyz \) satisfying the 3 conditions of the pumping lemma.

Because \( |xy| \leq p \), then \( xy \) only contains \( 0's \) and therefore \( y \) only contains \( 0's \). Since \( |y| > 0 \), it’s safe to claim that \( y = 0^k, k > 0 \). \( xy^0z = 0^{p-k}10^p \) which is not in the language \( A \). This contradicts the assumption that \( A \) is a regular language. Therefore, we conclude that \( A \) is not a regular language. \( \blacksquare \)

Proof of part c.

We assume that \( A \) is a regular language, then \( \overline{A} \) is also regular. There exists a pumping length \( p \). We choose the string \( s \) to be \( 0^p10^p \), which is in \( \overline{A} \) and the length of the string is \( 2p + 1 \geq p \). Thus, \( s \) can be divided into 3 pieces \( xyz \) satisfying the 3 conditions of the pumping lemma.

Because \( |xy| \leq p \), then \( xy \) only contains \( 0's \) and therefore \( y \) only contains \( 0's \). Since \( |y| > 0 \), it’s safe to claim that \( y = 0^k, k > 0 \). \( xy^0z = 0^{p-k}10^p \) which is not in the language \( \overline{A} \). This contradicts the assumption that \( \overline{A} \) is a regular language. Because \( \overline{A} \) is not a regular language, we conclude that \( A \) is not a regular language. \( \blacksquare \)
Problem 1.71 (2 pts)
Proof of part a.

For this problem, we just need to show that $A = 0(0 \cup 1)^*0$. At first, $0^k u 0^k$ is in $0(0 \cup 1)^*0$, therefore $A$ is a subset of $0(0 \cup 1)^*0$. For any $w$ in $0(0 \cup 1)^*0$, $w$ is in $A$ because $w = 0^t u 0^t$, so $0(0 \cup 1)^*0$ is a subset of $A$. ■

Proof of part b.

We assume that $B$ is a regular language, so there exists a pumping length $p$. We choose the string $s$ to be $0^p 110^p$, which is in $B$ and the length of the string is $2p + 2 \geq p$. Thus, $s$ can be divided into 3 pieces $xyz$ satisfying the 3 conditions of the pumping lemma.

Because $|xy| \leq p$, then $xy$ only contains 0’s and therefore $y$ only contains 0’s. Since $|y| > 0$, it’s safe to claim that $y = 0^k$, $k > 0$. $xy^2z = 0^{p+k} 110^p$ which is not in the language $B$. This contradicts the assumption that $B$ is a regular language. Therefore, we conclude that $B$ is not a regular language. ■
Problem 2.4  (2 pts each)

b) \[ S \rightarrow OT0 \mid 1T1 \]
   \[ T \rightarrow OT \mid 1T \mid \varepsilon \]

c) \[ S \rightarrow OT \mid 1T \]
   \[ T \rightarrow 0S \mid 1S \mid \varepsilon \]

e) \[ S \rightarrow 0SO \mid 1S1 \mid 0 \mid 1 \mid \varepsilon \]
Problem 2.9  (2 pts)

\[ S \rightarrow S_1 \mid S_2 \]
\[ S_1 \rightarrow S_1 c \mid T_1 \]
\[ S_2 \rightarrow a S_2 \mid T_2 \]
\[ T_1 \rightarrow a T_1 b \mid \epsilon \]
\[ T_2 \rightarrow b T_2 c \mid \epsilon \]

This is ambiguous.

Consider:

\[ S \]
\[ \downarrow \]
\[ S_1 \]
\[ \downarrow \]
\[ S_1 \]
\[ \downarrow \]
\[ S_1 \]
\[ \downarrow \]
\[ T_1 \]
\[ \downarrow \]
\[ a \]
\[ \downarrow \]
\[ a \]
\[ \downarrow \]
\[ a \]
\[ \downarrow \]
\[ a \]
\[ \downarrow \]
\[ a \]
\[ \downarrow \]
\[ b \]
\[ \downarrow \]
\[ c \]
\[ \downarrow \]
\[ abc \]

\[ S \]
\[ \downarrow \]
\[ S_2 \]
\[ \downarrow \]
\[ a \]
\[ \downarrow \]
\[ a \]
\[ \downarrow \]
\[ a \]
\[ \downarrow \]
\[ b \]
\[ \downarrow \]
\[ T_2 \]
\[ \downarrow \]
\[ b \]
\[ \downarrow \]
\[ T_2 \]
\[ \downarrow \]
\[ b \]
\[ \downarrow \]
\[ T_2 \]
\[ \downarrow \]
\[ b \]
\[ \downarrow \]
\[ c \]
\[ \downarrow \]
\[ abc \]

*Note: Alternative solutions are acceptable.*
Problem 2.10  (2 pts)

Verbal description on next page.

Diagram:

- Start
- $\epsilon, \epsilon \rightarrow \$\to 
- $\epsilon, \epsilon \rightarrow S\to 
- Loop with transitions:
  - $\epsilon, S_1 \rightarrow \epsilon, S_1\to 
  - $\epsilon, S_2 \rightarrow S_2\to 
  - $\epsilon, T_1 \rightarrow \epsilon, T_1\to 
  - $\epsilon, T_2 \rightarrow \epsilon, T_2\to 
- Accept

Production rules:

- $\epsilon, S \rightarrow S_1$
- $\epsilon, S \rightarrow S_2$
- $\epsilon, S_1 \rightarrow T_1$
- $\epsilon, S_1 \rightarrow \epsilon$
- $\epsilon, T_1 \rightarrow \epsilon$
- $\epsilon, T_2 \rightarrow \epsilon$
- $\epsilon, S_2 \rightarrow T_2$
- $\epsilon, S_2 \rightarrow \epsilon$
- $\epsilon, T_2 \rightarrow \epsilon$
- $a, a \rightarrow \epsilon$
- $b, b \rightarrow \epsilon$
- $c, c \rightarrow \epsilon$

\* Note: Alternative solutions are acceptable.
Problem 2.10 (2 pts)

PDA for A:
1. Nondeterministically jump to instruction 2 or 5
2. Read and push a’s
3. Read b’s while popping corresponding a’s
4. If b’s finish and the stack is empty, skip c’s on input and accept
5. Skip a’s on input
6. Read and push b’s
7. Read c’s while popping corresponding b’s
8. If c’s finish and the stack is empty accept