1. [5 pts. each] Describe a regular expression of the language accepted by each of the following deterministic finite state automaton.

(a) \( b^*a((a \cup b)a)^*(a \cup b)b^*a)^* = b^*a((a \cup b)b^*a)^* \)
Note that \( b^*a \) is for going from \( q_1 \) to \( q_2 \) and the rest is for going repeatedly from \( q_2 \) to \( q_2 \).

(b) \((ab^*a \cup bb \cup bab^*a)^*(ab^* \cup bab^* \cup b)\)
Note that \((ab^*a \cup bb \cup bab^*a)^*\) is for all possible combinations of going from 1 to 1 (1-2-1, 1-3-1, 1-3-2-1); \((ab^* \cup bab^* \cup b)\) is for finally reaching from 1 to 2 or to 3.

2. [5 pts. each] Give a deterministic finite state automaton accepting each of the following languages over the alphabet \( \{0, 1\} \). (You only need to show a state diagram.)

(a) All strings of length four or more in which the third symbol from the right end is different from the leftmost symbol.
(b) All strings in which the value of the string, interpreted as a binary representation of an integer, is zero modulo three. For example, 0110 and 1111, representing the integers 6 and 15, respectively, are to be accepted.

3. (5 pts. each)

(a) Give a deterministic finite state automaton accepting $L = (aa)^*(bb)^*b$.

(b) Convert the following NFA into an equivalent DFA.
4. (5 pts. each) Prove or disprove the following for regular expressions \( r \) and \( s \).

(a) \( (rs)^* = r^*s^* \).

No. Consider: \( r = a \) and \( s = b \). Then, \( abab \) is in \( (rs)^* \), but not in \( r^*s^* \).

(b) \( (r \cup s)^* = (r^*s^*)^* \).

First, note that \( (r \cup s) \subseteq r^*s^* \) implying that \( (r \cup s)^* \subseteq (r^*s^*)^* \). Next, we will prove that \( (r^*s^*)^* \subseteq (r \cup s)^* \). Clearly, \( r^*s^* \subseteq (r \cup s)^* \). This then implies that \( (r^*s^*)^* \subseteq ((r \cup s)^*)^* \). Note that \( ((r \cup s)^*)^* = (r \cup s)^* \). Therefore, \( (r \cup s)^* = (r^*s^*)^* \).

5. (5 pts. each) For each of the following languages, show that it is not regular. (Note that you may directly apply the Pumping lemma or may use the closure properties of the regular languages.)

(a) \( L = \{a^nb^n \mid n \geq 0 \} \)

Suppose \( L \) was regular, and consider a string \( w = a^pb^p \), where \( p \) is the Pumping length. \( w \) may then be split into three substrings \( w = xyz \), where \( x = a^\alpha \), \( y = a^\beta \), and \( z = a^\gamma b^p \) such that \( \alpha + \beta + \gamma = p^2 \), \( \beta \geq 1 \), and \( \alpha + \beta \leq p \). By the Pumping lemma, \( xy^2z \) must be then in \( L \), but the number of \( a \)'s in \( xy^2z \) cannot be a perfect square of any integer; hence, it is not in \( L \). Therefore, \( L \) is not regular.

(b) \( L = \{a^k b^l \mid k \neq l \} \) where \( k, l \geq 0 \)

Suppose \( L \) was regular, and consider a string \( w = a^{p(i + 1)}b^i \), where \( p \) is the Pumping length. \( w \) may then be decomposed into three substrings \( w = xyz \), where \( x = a^\alpha \), \( y = a^\beta \), and \( z = a^\gamma b^{p+1} \) such that \( \alpha + \beta + \gamma = p^2 \), \( \beta \geq 1 \), and \( \alpha + \beta \leq p \). Consider \( xy^i z \) where \( i = \frac{p(p+1)}{2} + 1 \). (Note that \( i \) is an integer.) Note that \( n_a(xy^i z) = p^2 + (i-1)\beta \) which is equal to \( (p+1)i \) which makes \( n_a(xy^i z) = n_b(xy^i z) \); hence, \( L \) cannot be regular.

(c) \( L = \{a \mid n_a(w) \neq n_b(w) \} \). Note that \( n_a(w) \) denotes the number of \( a \)'s in \( w \).

Suppose \( L \) was regular. Then, by choosing \( w = a^{p(i + 1)}b^i \), the same argument in part (b) can be applied to complete the proof.

6. (5 pts.) The nor of two languages is

\[ \text{nor}(L_1, L_2) = \{w \mid w \notin L_1 \text{ and } w \notin L_2\} \]

Show that the family of regular languages is closed under the nor operation.

Let \( L_1 \) and \( L_2 \) be two arbitrary regular languages. Then, \( \text{nor}(L_1, L_2) = \overline{L_1 \cup L_2} = \overline{L_1} \cap \overline{L_2} \).

Since regular languages are closed under the complement, both \( \overline{L_1} \) and \( \overline{L_2} \) are regular. As regular languages are also closed under the intersection, \( \overline{L_1} \cap \overline{L_2} \) must be regular as well which completes the proof.