There are a total of four problems. You have to solve **all** of them.

Problem 1 (CO1): DFA and Regular Languages (10 points)

Let $\Sigma = \{0, 1\}$. Consider the following languages over Σ.

$$L_1 = \{w : \text{the length of } w \text{ is at most three}\}$$

$$L_2 = \{w : w \text{ starts and ends with different letters}\}$$

$$L_3 = \{w : \text{the length of } w \text{ is at least two}\}$$

Now solve the following problems.

(a) Give the state diagram for a DFA that recognizes L_1. (2 points)

(b) Give the state diagram for a DFA that recognizes L_2. (3 points)

(c) Give the state diagram for a DFA that recognizes L_3. (2 points)

(d) Find a shortest string in $\overline{L_1} \cap L_3$. Here $\overline{L}$ denotes the complement of the language L i.e., $\overline{L} = \Sigma^* - L$. (1 point)

(e) If you were to use the "cross product" construction shown in class to obtain a DFA for the language $L_2 \cap L_3$, how many states would it have? (1 point)

(f) How many states does the smallest DFA for $L_2 \cap L_3$ have? (1 point)

Problem 2 (CO1): Regular Expressions (10 points)

Let $\Sigma = \{0, 1\}$. Consider the following pair of languages over Σ.

$$L_1 = \{w : \text{the length of } w \text{ is divisible by three}\}$$

$$L_2 = \{w : \text{every second letter in } w \text{ is a } 0\}$$

Now solve the following problems.

(a) Write down a regular expression for the language L_1. (2 points)

(b) Write down a regular expression for the language L_2. (2 points)

(c) Your friend wants a regular expression for the language $\overline{L_1 \cap L_2}$ where $\overline{L}$ denotes the complement of the language L i.e., $\overline{L} = \Sigma^* - L$. She wants your help. You tell her to make use of the fact $\overline{L_1 \cap L_2} = \overline{L_1} \cup \overline{L_2}$.

(i) Write down a regular expression for the language $\overline{L_1}$. (2 points)

(ii) Write down a regular expression for the language $\overline{L_2}$. (2 points)

(iii) Using the fact above, write down a regular expression for the language $\overline{L_1 \cap L_2}$. (2 points)

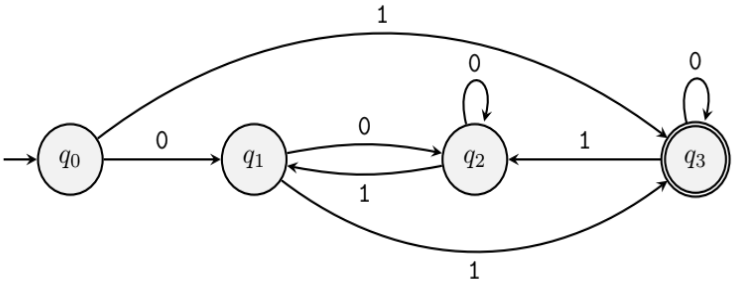
Problem 3 (CO1): Converting Regular Expressions to NFAs (10 points)

Convert the following regular expression over $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$ into an equivalent NFA. Note that $R_1 + R_2$ is the same as $R_1 \cup R_2$.

$$\mathtt{a}\mathtt{a}^* + \left(\mathtt{a} + \mathtt{b}^* \left(\mathtt{a} + \mathtt{c}\right)^*\right)^*$$

Problem 4 (CO1): Converting Finite Automata to Regular Expressions (10 points)

Convert the following DFA into an equivalent regular expression using the state elimination method. First eliminate q_2, then q_3 and finally q_1. You must show work.



After you are done with the test, please indicate where you stand on the smiley face spectrum.