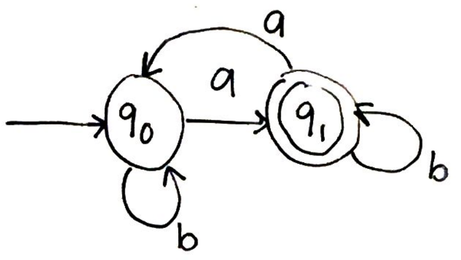
Constance Xu
I pledge my honor that I have abided by the Stevens Honor System.

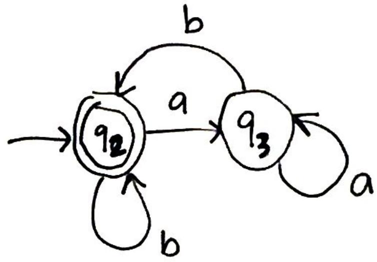
Problem 1A.

$L_1 = \{ w : w$ has an odd # of a's $\}$

$L_2 = \{ w : w$ ends w/ a b $\}$

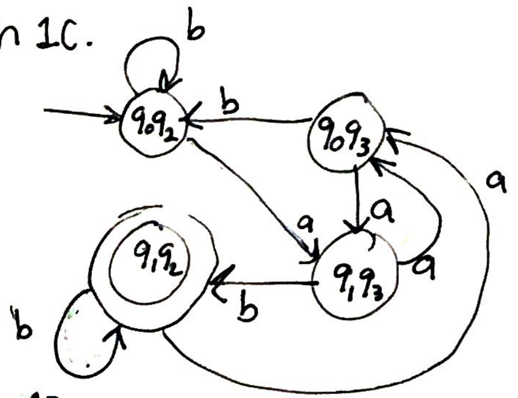
Problem 1B.





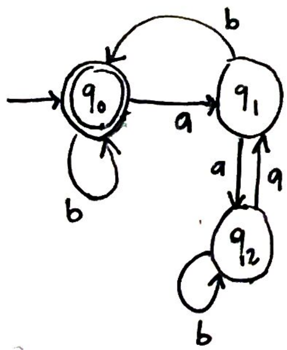
L_2

Problem 1C.



	A	B
q_0q_2	q_1q_3	q_0q_2
q_0q_3	q_1q_3	q_0q_2
q_1q_2	q_0q_3	q_1q_2
q_1q_3	q_0q_3	q_1q_2

Problem 1D.



I can merge the two states w/o changing the language because they are trying to accomplish the same things.

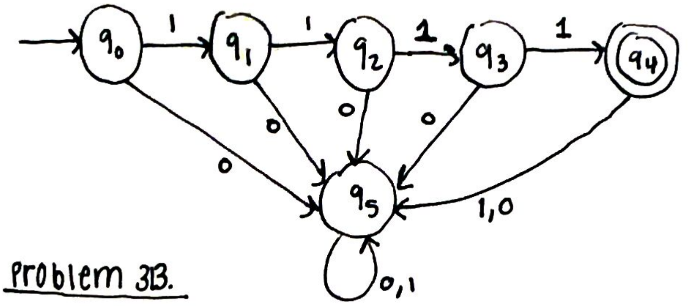
If you have 1 a, you still need a b. If you have 2 a's, the machine should NOT accept until you have 3 a's, then you'll need a b.

Problem 2

Start w/ an NFA X for A. Create this NFA as follows: If you reverse all the arrows for X (if they were going one direction, have them go the opposite way) and the start state of X should be the accept state for $X^{reversed}$.
In this NFA, create a new start state, q_0' and add ϵ transitions to each state of $X^{reversed}$ that correspond to the accept states of X. Make accept state q_{accept}
By doing so, we can show that for any string w in Σ, there is a path from the start state to the accept state... if and only if there is a path from q_0' to q_{accept}' in M!
Hence, $w \in A$ iff $w^R \in \cancel{A}X^{reversed}$

Problem 3A.

$L_4 = \{1111\}$



no, you cannot.

Problem 3B.

You cannot reduce the # of states because every state is completely necessary. By removing one, you completely change the language that the machine reads. Let a path from the start state to the accept state be of length k ($k \geq 3$). where

Where L_N is equal to N-ones. Let there be k amount of states where $N = k-2$. The -2 comes from the accept state and the initial state. We need one state for invalid input. This state is absolutely necessary and cannot be taken out.
If it were taken out, due to the pidgeon hole principle, they must be redirected to the other states, allowing for invalid strings to be validated.
Hence, you cannot take out any states.