1. Give DFA accepting the following languages $L \subseteq \{0,1\}^*$:

   (a) $L = \emptyset$

   (b) $L = \{\epsilon\}$

   (c) $L = \{w : 001 \text{ is a substring of } w\}$

   (d) $L = \{w : 001 \text{ is not a substring of } w\}$

   (e) $|w|_0$ be the number of 0's in the word w.

   $$L = \{w : |w|_0 = 3\}$$

   (f) Similarly $|w|_1$ be the number of 1's in the word w.

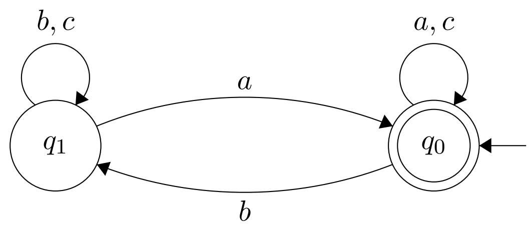
   $$L = \{w : |w|_0 \text{ is even and } |w|_1 \text{ is divisible by } 3\}$$

   (g) L be the set of strings that when interpreted as binary numbers, are divisible by 3.

   (h) L be the set of strings that when interpreted in *reverse* as binary numbers, are divisible by 3.

   Assume that leading zeroes are allowed in the binary numbers.

2. Consider the following automaton over the alphabet $\{a, c, b\}$. Is it a DFA?



   Try to identify the language it accepts, and formally prove the same.

3. Let M = (Q, Σ, δ, s, F) be an automaton. We defined the extended transition function $\hat{\delta}$ as follows:

   $$\hat{\delta}(q, \epsilon) = q \qquad \text{for all states } q \in Q$$
   $$\hat{\delta}(q, xa) = \delta(\hat{\delta}(q, x), a) \qquad \text{for all strings } x \in \Sigma^* \text{ and symbols } a \in \Sigma$$

   Prove the following:

   (a) For all states q and for all strings $x, y \in \Sigma^*$:

   $$\hat{\delta}(q, xy) = \hat{\delta}(\hat{\delta}(q, x), y)$$

   (b) For all states q, for all strings y and for all input symbols a:

   $$\hat{\delta}(q, ay) = \hat{\delta}(\delta(q, a), y)$$

4. Let $A = (Q, \Sigma, \delta, q_0, \{q_f\})$ be a DFA and suppose that for all $a \in \Sigma$, we have $\delta(q_0, a) = \delta(q_f, a)$.

   (a) Show that for all $w \neq \epsilon$, we have $\hat{\delta}(q_0, w) = \hat{\delta}(q_f, w)$.

(b) Show that if x is a nonempty string in $\mathcal{L}(A)$, then for all $k > 0, x^k$, i.e. x concatenated with itself k times, is also in $\mathcal{L}(A)$.

5. Show that if an NFA with k states accepts some word, then it accepts some word of length $k-1$ or less.

6. Given an alphabet $\Sigma = \{a_1, a_2, \cdots a_n\}$, construct an NFA that accepts exactly those words that do not contain all the letters from Σ, i.e. the language

$$\{w : \exists a_i \in \Sigma \text{ which does not appear in } w\}$$

Can you construct an NFA with at most n states that accepts the same language?

7. (a) Construct an NFA that accepts the language over $\{a, b\}$ of words that have third last letter a.

   (b) Use subset construction to obtain a DFA equivalent to the above NFA.

8. Construct an NFA that verifies addition of binary numbers. Suppose the problem is to add the numbers six and seven. then,

$$\begin{array}{r} 0110 \\ +0111 \\ --- \\ 1101 \end{array}$$

We shall encode this as a string on the alphabet

$$\Sigma = \left\{ \begin{bmatrix}0\\0\\1\end{bmatrix}, \begin{bmatrix}0\\1\\0\end{bmatrix}, \begin{bmatrix}0\\1\\1\end{bmatrix}, \begin{bmatrix}1\\1\\1\end{bmatrix}, \begin{bmatrix}1\\1\\0\end{bmatrix}, \begin{bmatrix}0\\0\\0\end{bmatrix}, \begin{bmatrix}1\\0\\0\end{bmatrix}, \begin{bmatrix}1\\0\\1\end{bmatrix} \right\}$$

where the first two rows represent the numbers to be added and the third row represents the sum. For instance, the above summation can be represented as the string:

$$\begin{bmatrix}0\\0\\1\end{bmatrix} \begin{bmatrix}1\\1\\1\end{bmatrix} \begin{bmatrix}1\\1\\0\end{bmatrix} \begin{bmatrix}0\\1\\1\end{bmatrix}$$

Construct an NFA that takes a string on the alphabet $\Sigma = M_{3\times1}(\{0,1\})$ (the set of three cross one matrices with zeros and ones as entries), and accepts the string if it represents a valid instance of addition.

How would you modify your automaton if the input was in decimal?