"Create with the heart, build with the mind." – Criss Jami

1. 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?

2. Let L, L_1, L_2 be languages over $\Sigma = \{a, b\}$. Recall from the lectures that $L_1L_2 = \{uv \mid u \in L_1, v \in L_2\}$ and $L_1^{-1}L_2 = \{v \mid \exists u \in L_1 \text{ such that } uv \in L_2\}$. State whether the following equations are true or false. Justify briefly.

   (a) $\{a\}^{-1}(\{a\}L) = L$

   (b) $\{a\}(\{a\}^{-1}L) = L$

   (c) $L_1^{-1}(L_1L_2) = L_2$

   (d) $L_1(L_1^{-1}L_2) = L_2$

   (e) $L^{-1}L = \{\varepsilon\}$

   (f) $L(L^{-1}L) = L$

3. For each part below, construct a complete DFA with at most 4 states that accepts at least each word in the left column, and rejects all words in the right column. It does not matter how your automaton behaves on other words.

(a)	$ababababa$ $bbbaaaa$ $abbbaab$	$aaabbb$ $babababa$ $aaababbb$
(b)	$aaabbaa$ $ababbab$ $abababab$	$bbbaabb$ $aaabbb$

4. 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 $\Sigma = M_{3\times 1}(\{0,1\})$ (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?