Formal Languages and Automata: NFA and Grammar Practice Problems

- **A.** Let $\Sigma = \{a, b, c\}$ and $L = \{w \in \Sigma^* | w \text{ ends in } cab \}$. Design an NFA for L.
- **B.** Let $\Sigma = \{a, b\}$ and let $L = \{w \in \Sigma^* | \text{ the third-from-last character of } w$ is $a \}$. Design an NFA for L. Your NFA should use at most four states.
- **C.** Let $\Sigma = \{a, b\}$ and $L = \{baa\}$. Design an NFA and DFA for L.
- **D.** Construct an NFA that recognizes strings over $\Sigma = \{a, b\}$ that have an odd number of a's.
- **E.** Construct an NFA that recognizes strings over the alphabet $\Sigma = \{a, b, c\}$ that start and end with the same symbol.

- **F.** Construct a regular grammar for the language of all strings over $\Sigma = \{a, b\}$ that start with 'a' and end with 'b'
- **G.** Construct a regular grammar for the language of all binary strings over the alphabet $\{0,1\}$ that have an even number of 1's.
- **I.** Construct a regular grammar for the language of all strings over $\Sigma = \{a, b\}$ that have at most one a.
- **J.** Construct a regular grammar for the language of all strings over $\Sigma = \{a, b\}$ where the substring abb appears.
- **K.** Construct a regular grammar for the language of all strings over $\Sigma = \{a, b\}$ that are palindromes.
- **L.** Construct a regular grammar for the language of all strings over $\Sigma = \{a, b\}$ that do not contain the substring aa.