Problem 1. [Category: Design] Construct a DFA for the language \( L = L((01 + 10)^*(\epsilon + 0 + 1)) \) by the following steps.

1. Construct an NFA for \( L((01 + 10)^*(\epsilon + 0 + 1)) \) by following the inductive translation from regular expressions to NFAs.
2. Convert the NFA constructed in step 1 into a DFA. Construct only the reachable part of the automaton.
3. Complement the DFA in step 2 obtain a DFA for language \( L \).

Problem 2. [Category: Design] Fix some alphabet \( \Sigma \). For a given language \( L \subseteq \Sigma^* \), define the following three languages.

- \( \text{PREFIX}(L) = \{ u \in \Sigma^* | \exists w \in \Sigma^* \text{ such that } uw \in L \} \)
- \( \text{SUFFIX}(L) = \{ u \in \Sigma^* | \exists w \in \Sigma^* \text{ such that } wu \in L \} \)
- \( \text{MID}(L) = \{ y | \exists x, z \in \Sigma^* \text{ such that } xyz \in L \} \)

Show that \( \text{PREFIX}(L), \text{SUFFIX}(L) \) and \( \text{MID}(L) \) are regular if \( L \) is regular. A useful technique here is to construct an NFA \( N \) that accepts each of these languages assuming that there is a DFA \( M \) that accepts \( L \). More concretely, assume \( M = (Q, \Sigma, \delta, s, A) \). Describe an NFA \( N = (Q', \Sigma, \delta', s', A') \) where each of \( Q', \delta', s', A' \) are defined in terms of \( Q, \delta, s, A \) and potentially some additional information.