3. Regular languages, Finite Automata

Exercise 3.2 can be made and handed in; Exercise (*) is a bit harder. Let \( \Sigma = \{a, b\} \).

3.1. 1. Construct a DFA \( M_1 \) such that
\[
L(M) = L_1 = \{ w \in \Sigma^* \mid \#_a(w) \text{ is divisible by 3} \}.
\]

2. Construct an \( M_2 \) such that
\[
L(M) = L_2 = \{ w \in \Sigma^* \mid \#_b(w) \text{ is divisible by 2} \}.
\]

3. Construct a NFA \( \lambda \) such that \( L(M) = L_1 \cup L_2 \).

4. Construct a DFA \( M_4 \) such that \( L(M) = L_1 \cup L_2 \).

3.2. Construct a regular expression \( e \) such that
\[
L(e) = L = \{ w \in \Sigma^* \mid \text{‘abba’ does not occur in } w \}.
\]

We do this in several steps
1. First find an NFA \( M \) such that its language is \( \overline{L} \).
2. Then construct a DFA \( M’ \) accepting the same language.
3. Modify \( M’ \) to obtain \( M” \) accepting \( L \).
4. Find \( e \) such that \( L(e) = L(M”) \).

3.3. Assume that \( L \) over \( \Sigma \) is regular.

1. Show that \( L_1 := \{ w \in \Sigma^* \mid \exists v \in L(w \text{ is an initial segment of } v) \} \) is regular.

2. (*) Show that \( L_2 := \{ w \in \Sigma^* \mid \exists v \in L(w \text{ is contained in } v) \} \) is regular.

NB \( w \) is an initial segment of \( v \) if \( v = wu \) for some \( u \);
\( w \) is contained in \( v \) if all the symbols of \( w \) occur in \( v \) in that order, to be precise:
\( v = s_1 \ldots s_n \) and \( w = s_{i_1} \ldots s_{i_m} \) for some sequence \( 1 \leq i_1 < \ldots < i_m \leq n. \ (s_j \in \Sigma) \)