Exercise 1: Constructing DFAs

Construct DFAs that recognize the following languages. Drawing the state diagrams is sufficient. The alphabet is \( \Sigma = \{0, 1\} \).

(a) \( L_1 = \{w \mid |w| \geq 2 \text{ and } w \text{ contains an odd number of ones}\} \).
(b) \( L_2 = \{w \mid w \text{ contains an even number of zeros}\} \).
(c) \( L_3 = \{w \mid w \text{ every zero is immediately followed by a one}\} \).
(d) \( L_4 = \{w \mid w \text{ ends with } 01\} \).

Exercise 2: Kleene Star

For two languages \( L \) and \( K \) we define

\[
L \cdot K := \{ww' \mid w \in L, w' \in K\}.
\]

Use counter examples to show that none of the following equalities is valid for general languages \( K \) and \( L \).

1. \( K \cdot L = L \cdot K \)
2. \( (K \cdot K)^* = K^* \)
3. \( K^* \cdot K = K^* \)
4. \( (LK)^* = (L \cup K)^* \)
Exercise 3: From NFA to DFA (1+2+2 Points)

Consider the following NFA.

(a) Give a formal description of the NFA by giving the alphabet, state set, transition function, start state and the set of accept states.

(b) Construct a DFA which is equivalent to the above NFA by drawing the corresponding state diagram.

(c) Explain which language the automaton accepts.

Exercise 4: Union of regular Languages (3 Points)

Let $L_1, L_2$ be regular languages. Show that $L_1 \cup L_2$ is also regular without (explicitly!) using the proof which was presented in the lecture.