1 DFA Proofs

Problem. Give a DFA that accepts the language $\{x \in \{0,1\}^* \mid \text{ the second last letter of } x \text{ is } 1\} = \mathfrak{L}((0+1)^*1(0+1)$.

Solution.

For $\delta^*(q_{00}, \lambda) q_{00} \notin F$, $\delta^*(q_{00}, 1) = q_{01} \notin F$, $\delta^*(q_{00}, 0) = q_{00} \notin F$, we can prove by induction that

$$\begin{aligned}
&\forall n \in \mathbb{N}.(n \geq 2 \text{ IMPLIES} && (1)\\
&(\forall x \in \{0,1\}^n. && (2)\\
&[(\delta^*(q_{00}, x) = q_{00} \text{ IMPLIES 00 is a suffix of x} && (3)\\
&\delta^*(q_{00}, x) = q_{01} \text{ IMPLIES 01 is a suffix of x} && (4)\\
&\delta^*(q_{00}, x) = q_{10} \text{ IMPLIES 10 is a suffix of x} && (5)\\
&\delta^*(q_{00}, x) = q_{11} \text{ IMPLIES 11 is a suffix of x])) } && (6)
\end{aligned}$$

We can start as follows.

Let $n \geq 2$.

$\forall x \in \{0,1\}^n.\delta^*(q_{00}, x) \in F$ IFF the second last letter of x is 1.

□

2 Nondetermenistic Finite Automata (NFA)

An NFA is a 5-tuple $(Q, \Sigma, \delta, q_0, f)$, wher Q is a finite set of states, Σ is a finite alphabet, $q_0 \in Q$ is an initial state, $F \subseteq Q$, and $\delta : Q \times \Sigma \to 2^Q$.

Then we define $\delta^*(q, \lambda) = \{\lambda\}$.

For all $a \in \Sigma$, $x \in \Sigma^*$, $\delta^*(q, xa) = \bigcup\{\delta(q', a) \mid q' \in \delta^*(q, x)\}$, or, equivalently, $\delta^*(q, ax) = \bigcup\{\delta^*(q', x) \mid q' \in \delta(q, a)\}$.

A string is accepted by a DFA if the path labelled by x starting from q_0 ends in a final stat.

x is accepted by a NFA if there exists a path labelled by x starting from q_0 that ends in a final state $L(M) = \{x \in \Sigma^* \mid \delta^*(q_0, x) \cap F \neq \emptyset\}$.

Every DFA $(Q, \Sigma, \delta, q_0, F$ can be easily transformed to an NFA $(Q, \Sigma, \delta, q_0, F)$ by defining $\gamma(q, a) = \{\delta(q, a)\}$.

Question. Are there some languages that can be accepted by an NFA but not by a DFA?

3 Subset Construction

Theorem 3.1

For every NFA $M = (Q, \Sigma, \delta, q_0, F)$ there is a DFA $\widehat{M} = (\widehat{Q}, \Sigma, \widehat{\delta}, \widehat{q_0}, \widehat{F})$ such that $\mathfrak{L}(M) = \mathfrak{L}(\widehat{M})$.

Proof.

Let $M = (Q, \Sigma, \delta, q_0, F)$ be an arbitrary NFA.

The DFA we construct keeps trakn of the possible states in which M can be as it reads the input string.

Let $\widehat{M} = (\widehat{Q}, \Sigma, \widehat{\delta}, \widehat{q_0}, \widehat{F})$, where $\widehat{Q} = 2^Q$, $\widehat{q_0} = \{q_0\}$. Note that $\widehat{q_0} \in \widehat{Q}$.

For $Q' \in 2^Q$ and $a \in \Sigma$, $\widehat{\delta}(Q', a) = \bigcup\{\delta(q, a) \mid q \in Q'\}$.

Let $\widehat{F} = \{Q' \in 2^Q \mid Q' \cap F = \emptyset\}$. $\square$

Claim. $\mathfrak{L}(M) = \mathfrak{L}(\widehat{M})$.

Proof.

For all $w \in \Sigma^*$, let $Q(w) =$ "$\widehat{\delta}^*(\{q_0\}, w) = \delta^*(q_0, w)$".

$\forall w \in \ S^*.P(w)$ by induction on the length of w or structural induction on the string. $\square$