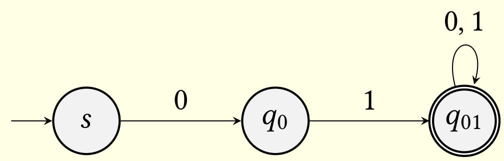
1. For the following languages, construct a DFA accepting the language and prove that your construction is correct.

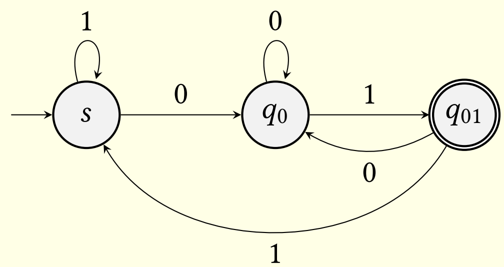
   (a) $L = \{w \in \{0,1\}^* \mid w \text{ starts with } 01\}$.

   Solution:

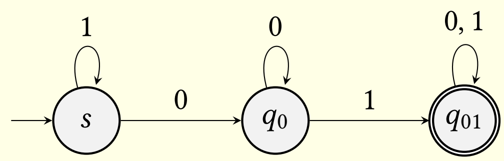
   (b) $L = \{w \in \{0,1\}^* \mid w \text{ ends with } 01\}$.

   Solution:

   (c) $L = \{w \in \{0,1\}^* \mid w \text{ contains } 01 \text{ as a substring}\}$.

   Solution:

   Let's try to argue why this construction is correct. Firstly, the set of strings w such that $\widehat{\delta}(s,w) = s$ is $\{1^n \mid n \in \mathbb{N}\}$. Similarly, the set of strings w such that $\widehat{\delta}(s,w) = q_0$ iff w is a sequence of 1s followed by a sequence of 0s. This is because the only incoming transitions to q_0 are from s and q_0, and both are labelled by 0. Similarly, if the DFA enters q_{01}, then the previous state was q_0 (that is the only way to reach q_{01}). Hence for every string w, $\widehat{\delta}(s,w) = q_{01}$ iff w contains 01 as a substring.

   (d) $L = \{w \in \{0,1\}^* \mid \text{every block of four symbols in } w \text{ contains at least two 0s}\}$.

   Solution: Let's look at $\overline{L}$ and then split that into simpler languages, and use the closure properties.

   $$\overline{L} = \{w \mid w \text{ contains a substring that contains less than two 0s}\}.$$

   Now, $\overline{L}$ can be written as the union of L_{1111}, L_{0111}, L_{1011}, L_{1101}, and L_{1110} where the language L_{abcd} is defined as follows.

   $$L_{abcd} = \{w \mid w \text{ contains } abcd \text{ as a substring}\}.$$

For L_{abcd}, you can do a construction similar to the (c)-part of this question. You will also be able to reason about the correctness of L_{abcd} in a similar way. The language L is now obtained by the closure properties.

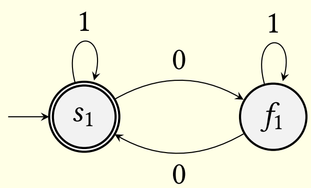
A brief remark: This need not be the DFA with the least number of states, but it is definitely one where you can reason about the correctness easily. Sometimes, it is alright to trade-off optimality for the ease of proving correctness.

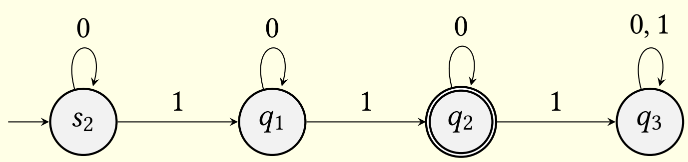
(e) $L = \{w \in \{0,1\}^* \mid w \text{ contains an even number of 0s or contains exactly two 1s}\}$.

Solution: Once again the easiest solution is to use the union of

$$L_1 = \{w \in \{0,1\}^* \mid w \text{ contains an even number of 0s}\},$$
$$L_2 = \{w \in \{0,1\}^* \mid w \text{ contains exactly two 1s}\}.$$

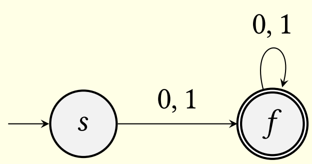
The corresponding DFAs are as follows.





(f) $L = \{0,1\}^* \setminus \{\epsilon\}$.

Solution:



For these languages, argue that your construction is correct. First, argue with normal English sentences. Now, try to formalize this by writing the correct induction hypothesis like we did in class.

2. Let Σ be a finite alphabet and let $x, y \in \Sigma^*$ be any two strings. Show that $xy = yx$ iff there exists a string $z \in \Sigma^*$ such that $x^2y^2 = z^2$.

Solution: We will prove both directions of the iff statement, starting with the easier direction.

($\Rightarrow$) If $xy = yx$, then $x^2y^2 = xxyy = xyxy$. Hence there is a $z = xy$ such that $x^2y^2 = z^2$.

($\Leftarrow$) Let $z \in \Sigma^*$ such that $x^2y^2 = z^2$. Note that if $|x| = |y|$ then $x = y$ and we are done. Without loss of generality, assume that $|x| > |y|$. We let $x = x_1x_2$ so that $|x_1| = |y|$.

Therefore, $x^2y^2 = z^2 \implies xxyy = z^2 \implies x_1x_2x_1x_2yy = z^2$. Observe that $|x_1x_2x_1| = |x_2yy|$. Since $x_1x_2x_1x_2yy = zz$, this implies that $x_1x_2x_1 = x_2yy = z$. Since $|x_1| = |y|$, $x_1x_2 = x_2y$ and $x_1 = y$. Therefore, $x_1x_2x_1x_2yy = z^2 \implies x_1x_2yx_1x_2y = z^2$ i.e., $xyxy = xxyy \implies xy = yx$.

3. Let $\Sigma = \{0, 1\}^*$ and $L \subseteq \Sigma^*$ be the set strings w such that the number of 0s in w is not equal to the number of 1s in w. Can you describe the language L^2? Is it regular?

Solution: We will show that $\overline{L^2}$ is regular (via a DFA) and since regular languages are closed under complementation, it follows that L^2 is regular.

Define $\#0(z)$ and $\#1(z)$ to be the number of 0s and 1s present in string z resp. Let the predicate Eqv be defined as $\mathrm{Eqv}(z) \stackrel{\text{def}}{=} (\#0(z) = \#1(z))$. From the definition of $\overline{L^2}$, we can say that $x \in \overline{L^2} \Leftrightarrow \forall y$ and z, $(x = yz) \Rightarrow (\mathrm{Eqv}(y) \bigvee \mathrm{Eqv}(z))$.

Claim 1. *$x \in \overline{L^2} \Leftrightarrow x$ starts and ends with the same symbol, and no two consecutive symbols are identical.*

In other words, the strings in $\overline{L^2}$ will be 1 followed by a series of 01s or 0 followd by a series of 10s.

Proof of Claim 1. We will prove the two directions of the iff statement.

($\Leftarrow$) Any string x that has the same starting and ending symbols, and no two consecutive symbols that are same will have $|\#0(x) - \#1(x)| = 1$. Thus no matter how you split this string, at least one of the parts will have the same number of 0s and 1. Consequently, $x \notin L^2$.

($\Rightarrow$) Firstly, observe that the string x must be of odd length, otherwise x can be expressed as the concatenation of two odd length strings y, z. For odd length strings w, $\mathrm{Eqv}(w)$ is false and this would imply that $x \in L^2$.

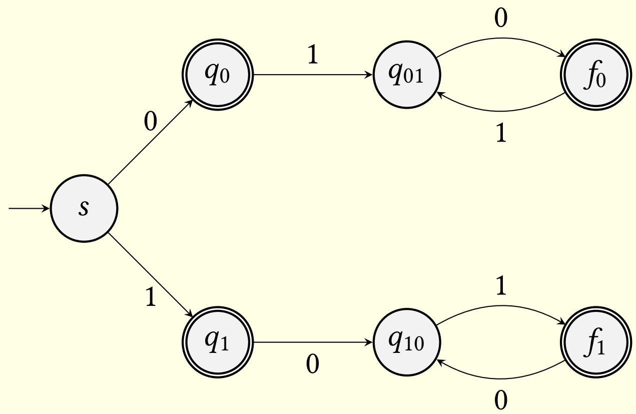
Now, let $x = x_0x_1x_2 \ldots x_n \in \overline{L^2}$. Consider the partition $P_0 = ((x_0), (x_1x_2 \ldots x_n))$. Clearly, $\mathrm{Eqv}(x_0)$ is false and therefore $\mathrm{Eqv}(x_1x_2 \ldots x_n)$ must hold. For the next partition $P_1 = ((x_0x_1), (x_2 \ldots x_n))$, $\mathrm{Eqv}(x_2 \ldots x_n)$ cannot hold (we are removing one bit from a balanced string) and therefore $\mathrm{Eqv}(x_0x_1)$ holds. This implies that $x_0 = \overline{x_1}$. In partition $P_3 = ((x_0x_1x_2), (x_3 \ldots . x_n))$, $\mathrm{Eqv}(x_3x_4 \ldots x_n)$ must hold since $\mathrm{Eqv}(x_0x_1x_2)$ does not hold and $x \notin L^2$. Thus, it must be the case that $x_2 = \overline{x_1}$, and hence $x_2 = x_0$.

By extending the argument for partitions P_i in general (P_i consists of the partition where $x_0\ x_1 \ldots x_i$ is in the first part), We can infer that all bits in even positions are of same parity and all bits in odd positions are of opposite parity.

Combined with the observation that the strings in $\overline{L^2}$ are of odd length, this completes the proof of Claim 1.

□

$\overline{L^2}$ is regular via the following DFA M.



There are certain transitions missing in this DFA - for instance the transition from q_0 on a 0. You can assume that all such transitions move to a dead state from which M does not escape.