Let L be an arbitrary regular language over the alphabet $\Sigma = \{0, 1\}$. Prove that the following languages are also regular. (You probably won't get to all of these during the lab session.)

1. Let $\textsc{InsertAny1s}(L)$ is the set of all strings that can be obtained from strings in L by inserting ***any number of*** 1s anywhere in the string. For example:

   $$\textsc{InsertAny1s}(\{\varepsilon,\ 1,\ 00\}) = \{\varepsilon,\ 1,\ 11,\ 111, \ldots,\ 00,\ 100,\ 0111110, 1110111111101111, \ldots\}$$

   Prove that the language $\textsc{InsertAny1s}(L)$ is regular.

   Solution: Let $M = (Q, s, A, \delta)$ be an arbitrary DFA that accepts the regular language L. We construct a new ***NFA with ε-transitions*** $M' = (Q', s', A', \delta')$ that accepts $\textsc{InsertAny1s}(L)$ as follows.

   Intuitively, M' *guesses* which 1s in the input string have been inserted, skips over those 1s, and simulates M on the original string w. M' has the same states and start state and accepting states as M, but it has a different transition function.

   $$Q' = Q$$
   $$s' = s$$
   $$A' = A$$

   $$\delta'(q, 0) = \{\ \delta(q, 0)\ \}$$
   $$\delta'(q, 1) = \{\qquad\qquad\qquad\qquad\qquad\qquad\}$$
   $$\delta'(q, \varepsilon) = \{\qquad\qquad\qquad\qquad\qquad\qquad\}$$

   ■

2. Let DELETEANY1s(L) is the set of all strings that can be obtained from strings in L by inserting ***any number of*** 1s anywhere in the string. For example:

$$\text{DELETEANY1s}(\{\varepsilon,\ 00,\ 1101\}) = \{\varepsilon, 0, 00, 01, 10, 101, 110, 1101\}$$

Prove that the language DELETEANY1s(L) is regular.

Solution: Let $M = (Q, s, A, \delta)$ be an arbitrary DFA that accepts the regular language L. We construct a new ***NFA with ε-transitions*** $M' = (Q', s', A', \delta')$ that accepts $DeleteAny1s(L)$ as follows.

Intuitively, M' *guesses* where 1s have been deleted from its input string, and simulates the original machine M on the guessed mixture of input symbols and 1s. M' has the same states and start state and accepting states as M, but a different transition function.

$$Q' = Q$$
$$s' = s$$
$$A' = A$$

$$\delta'(q, 0) = \{\ \delta(q, 0)\ \}$$
$$\delta'(q, 1) = \{\qquad\qquad\qquad\qquad\qquad\qquad\}$$
$$\delta'(q, \varepsilon) = \{\qquad\qquad\qquad\qquad\qquad\qquad\}$$

■

3. Let $\textsc{InsertOne1}(L) := \{x\mathtt{1}y \mid xy \in L\}$ denote the set of all strings that can be obtained from strings in L by inserting ***exactly one*** `1`. For example:

$$\textsc{InsertOne1}(\{\varepsilon, \mathtt{00}, \mathtt{101101}\}) = \{\mathtt{1}, \mathtt{100}, \mathtt{010}, \mathtt{001}, \mathtt{1101101}, \mathtt{1011101}, \mathtt{1011011}\}$$

Prove that the language $\textsc{InsertOne1}(L)$ is regular.

Solution: Let $M = (Q, s, A, \delta)$ be an arbitrary DFA that accepts the regular language L. We construct a new ***NFA with ε-transitions*** $M' = (Q', s', A', \delta')$ that accepts $\textsc{InsertOne1}(L)$ as follows.

If the input string w does not contain a `1`, then M' must rejects it; otherwise, intuitively, M' *guesses* which `1` was inserted into w, skips over that `1`, and simulates M on the remaining string xy.

M' consists of two copies of M, one to process the prefix x and the other to process the suffix y. State $(q, \textsc{False})$ means (the simulation of) M is in state q and M' has not yet skipped over a `1`. State $(q, \textsc{True})$ means (the simulation of) M is in state q and M' has already skipped over a `1`.

$$\begin{aligned}
Q' &= Q \times \{\textsc{True}, \textsc{False}\} \\
s' &= (s, \textsc{False}) \\
A' &=
\end{aligned}$$

$$\begin{aligned}
\delta'((q, \textsc{False}), \mathtt{0}) &= \{\, (\delta(q, \mathtt{0}), \textsc{False}) \,\} \\
\delta'((q, \textsc{False}), \mathtt{1}) &= \{ \qquad\qquad\qquad\qquad \} \\
\delta'((q, \textsc{False}), \varepsilon) &= \{ \qquad\qquad\qquad\qquad \} \\
\delta'((q, \textsc{True}), \mathtt{0}) &= \{ \qquad\qquad\qquad\qquad \} \\
\delta'((q, \textsc{True}), \mathtt{1}) &= \{ \qquad\qquad\qquad\qquad \} \\
\delta'((q, \textsc{True}), \varepsilon) &= \{ \qquad\qquad\qquad\qquad \}
\end{aligned}$$

■

4. Let $\textsc{DeleteOne1}(L) := \{xy \mid x\mathtt{1}y \in L\}$ denote the set of all strings that can be obtained from strings in L by deleting exactly one $\mathtt{1}$. For example:

$$\textsc{DeleteOne1}(\{\varepsilon,\ \mathtt{00},\ \mathtt{101101}\}) = \{\mathtt{01101},\ \mathtt{10101},\ \mathtt{10110}\}$$

Prove that the language $\textsc{DeleteOne1}(L)$ is regular.

Solution: Let $M = (\Sigma, Q, s, A, \delta)$ be a DFA that accepts the regular language L. We construct an ***NFA with ε-transitions*** $M' = (\Sigma, Q', s', A', \delta')$ that accepts $\textsc{DeleteOne1}(L)$ as follows.

Intuitively, M' *guesses* where the $\mathtt{1}$ was deleted from its input string. It simulates the original DFA M on the prefix x before the missing $\mathtt{1}$, then the missing $\mathtt{1}$, and finally the suffix y after the missing $\mathtt{1}$.

M' consists of two copies of M, one to process the prefix x and the other to process the suffix y. State $(q, \textsc{False})$ means (the simulation of) M is in state q and M' has not yet reinserted a $\mathtt{1}$. State $(q, \textsc{True})$ means (the simulation of) M is in state q and M' has already reinserted a $\mathtt{1}$.

$$Q' = Q \times \{\textsc{True}, \textsc{False}\}$$
$$s' = (s, \textsc{False})$$
$$A' =$$

$$\delta'((q, \textsc{False}), \mathtt{0}) = \{\ (\delta(q, \mathtt{0}), \textsc{False})\ \}$$
$$\delta'((q, \textsc{False}), \mathtt{1}) = \{\qquad\qquad\qquad\qquad\}$$
$$\delta'((q, \textsc{False}), \varepsilon) = \{\qquad\qquad\qquad\qquad\}$$

$$\delta'((q, \textsc{True}), \mathtt{0}) = \{\qquad\qquad\qquad\qquad\}$$
$$\delta'((q, \textsc{True}), \mathtt{1}) = \{\qquad\qquad\qquad\qquad\}$$
$$\delta'((q, \textsc{True}), \varepsilon) = \{\qquad\qquad\qquad\qquad\}$$

■

Work on these later: Consider the following recursively defined function on strings:

$$evens(w) := \begin{cases} \varepsilon & \text{if } w = \varepsilon \\ \varepsilon & \text{if } w = a \text{ for some symbol } a \\ b \cdot evens(x) & \text{if } w = abx \text{ for some symbols } a \text{ and } b \text{ and some string } x \end{cases}$$

Intuitively, $evens(w)$ skips over every other symbol in w, starting with the first symbol. For example, $evens(\texttt{THE}\diamond\texttt{SNAIL}) = \texttt{H}\diamond\texttt{NI}$ and $evens(\texttt{GROB}\diamond\texttt{GOB}\diamond\texttt{GLOB}\diamond\texttt{GROD}) = \texttt{RBGBGO}\diamond\texttt{RD}$.

Let L be an arbitrary regular language over the alphabet $\Sigma = \{\texttt{0}, \texttt{1}\}$.

5. Prove that the language $\textsc{Unevens}(L) := \{w \mid evens(w) \in L\}$ is regular.

6. Prove that the language $\textsc{Evens}(L) := \{evens(w) \mid w \in L\}$ is regular..