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. $\text{FlipOdds}(L) := \{\mathit{flipOdds}(w) \mid w \in L\}$, where the function *flipOdds* inverts every odd-indexed bit in w. For example:

$$\mathit{flipOdds}(\underline{0}0\underline{0}0\underline{1}1\underline{1}1\underline{0}1\underline{0}1\underline{0}1\underline{0}0) = \underline{1}0\underline{1}0\underline{0}1\underline{0}1\underline{1}1\underline{1}1\underline{1}1\underline{1}0$$

Solution: Let $M = (Q, s, A, \delta)$ be an arbitrary DFA that accepts L. We construct a new **DFA** $M' = (Q', s', A', \delta')$ that accepts $\text{FlipOdds}(L)$ as follows.

Intuitively, M' receives some string $\mathit{flipOdds}(w)$ as input, restores every other bit to obtain w, and simulates M on the restored string w.

Each state (q, flip) of M' indicates that M is in state q, and we need to flip the next input bit if $\mathit{flip} = \text{True}$.

$$Q' = Q \times \{\text{True}, \text{False}\}$$
$$s' = (s, \text{True})$$
$$A' =$$
$$\delta'((q, \mathit{flip}), a) =$$

■

2. UNFLIPODD1s(L) := $\{w \in \Sigma^* \mid \mathit{flipOdd1s}(w) \in L\}$, where the function *flipOdd1* inverts every other 1 bit of its input string, starting with the first 1. For example:

$$\mathit{flipOdd1s}(0000\underline{1}1\underline{1}100\underline{1}01\underline{0}1\underline{0}) = 0000\underline{0}1\underline{0}100\underline{0}01\underline{0}0\underline{0}$$

Solution: Let $M = (Q, s, A, \delta)$ be an arbitrary DFA that accepts L. We construct a new **DFA** $M' = (Q', s', A', \delta')$ that accepts UNFLIPODD1s(L) as follows.

Intuitively, M' receives some string w as input, flips every other 1 bit, and then simulates M on the transformed string.

Each state (q, flip) of M' indicates that M is in state q, and we need to flip the next 1 bit if and only if $\mathit{flip} = \text{TRUE}$.

$$Q' = Q \times \{\text{TRUE}, \text{FALSE}\}$$
$$s' = (s, \text{TRUE})$$
$$A' =$$
$$\delta'((q, \mathit{flip}), a) =$$

■

3. FLIPODD1s(L) := $\{flipOdd1s(w) \mid w \in L\}$, where the function *flipOdd1* is defined as in the previous problem.

Solution: Let $M = (Q, s, A, \delta)$ be an arbitrary DFA that accepts L. We construct a new **NFA** $M' = (Q', s', A', \delta')$ that accepts FLIPODD1s(L) as follows.

Intuitively, M' receives some string $flipOdd1s(w)$ as input, ***guesses*** which 0 bits to restore to 1s, and simulates M on the restored string w. No string in FLIPODD1s(L) has two 1s in a row, so if M' ever sees 11, it must reject.

Each state $(q, flip)$ of M' indicates that M is in state q, and we need to flip some 0 bit before the next 1 bit if $flip = \text{TRUE}$.

$$
\begin{aligned}
Q' &= Q \times \{\text{TRUE}, \text{FALSE}\} \\
s' &= (s, \text{TRUE}) \\
A' &= \\
\delta'((q, flip), a) &=
\end{aligned}
$$

■

4. $\textsc{Shuffle}(L) := \{ \mathit{shuffle}(w, x) \mid w, x \in L \text{ and } |w| = |x| \}$, where the function *shuffle* is defined recursively as follows:

$$\mathit{shuffle}(w, x) := \begin{cases} x & \text{if } w = \varepsilon \\ a \cdot \mathit{shuffle}(x, y) & \text{if } w = ay \text{ for some } a \in \Sigma \text{ and some } y \in \Sigma^* \end{cases}$$

For example, $\mathit{shuffle}(\texttt{0001101}, \texttt{1111001}) = \texttt{01010111100011}$.

Solution: Let $M = (Q, s, A, \delta)$ be an arbitrary DFA that accepts L. We construct a new **DFA** $M' = (Q', s', A', \delta')$ that accepts $\textsc{Shuffle}(L)$ as follows.

Intuitively, M' reads the string $\mathit{shuffle}(w, x)$ as input, splits the string into the subsequences w and x, and passes those strings to two independent copies of M. Let M_1 denote the copy that processes the first string w, and let M_2 denote the copy that processes the second string x.

Each state $(q_1, q_2, \mathit{next})$ indicates that machine M_1 is in state q_1, machine M_2 is in state q_2, and *next* indicates whether M_1 or M_2 receives the next input bit.

$$Q' = Q \times Q \times \{1, 2\}$$
$$s' = (s, s, 1)$$
$$A' =$$
$$\delta'((q_1, q_2, \mathit{next}), a) =$$

■