Problem 1. Give state diagrams of DFAs recognizing the following languages. In all cases
the alphabet is \{0, 1\}.

(a) \{w \mid w \text{ begins with a 1 and ends with a 0}\}.

(b) \{w \mid \text{starts with 0 and has odd length, or starts with 1 and has even length}\}.

(c) \{w \mid w \text{ doesn’t contain the substring 110}\}.

(d) \{w \mid w \text{ every odd position of } w \text{ is a 1}\}.

(e) \{w \mid w \text{ contains at least two 0s and at most one 1}\}.

Problem 2. A deterministic finite-state transducer is a device much like a DFA, except
that its purpose is not to accept strings or languages but to transform input strings into
output strings. Informally, it starts in a designated initial state and moves from state to
state, depending on the input, just as a DFA does. On each step, however, it emits a string
of zero or one or more symbols, depending on the current state and the input symbol. The
state diagram for a deterministic finite-state transducer looks like that for a DFA, except
that the label on an arrow looks like \(a/w\), which means “if the input symbol is \(a\), follow
this arrow and output \(w\)”. For example, the deterministic finite-state transducer over \(\{a, b\}\)
shown below outputs all \(b\)’s in the input string but omits every other \(a\).

\[
\begin{array}{c}
q_1 \\
\searrow \downarrow \uparrow \\
\bullet \\
\nearrow \downarrow \uparrow \\
q_2 \\
\end{array}
\]

\(b/b\) \quad \(a/a\) \quad \(b/b\)

\(a/\varepsilon\)

Draw state diagrams for deterministic finite-state transducers over \(\{a, b\}\) that do the
following.

(a) On input \(w\), outputs an \(a\) for each occurrence of the substring \(ab\) in \(w\).

(b) On input \(w\), outputs an \(a\) for each occurrence of the substring \(aba\) in \(w\).
(c) On input $w$, produce a string of length $|w|$ whose $i$th symbol is an $a$ if $i = 1$, or if $i > 1$ and the $i$th and $(i - 1)$st symbols of $w$ are different; otherwise, the $i$th symbol of the output is a $b$. For example, on input $aabba$ the transducer should output $ababa$, and on input $aaaab$ it should output $abba$.

**Problem 3.** Let $M$ be a DFA. Under exactly what circumstances is $\varepsilon \in L(M)$? Explain why that is the case using the notions of automata computation and acceptance.

**Problem 4.** Consider the following alphabet

$$\Sigma_3 = \left\{ \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \ldots, \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \right\}. $$

$\Sigma_3$ contains all size 3 columns of 0s and 1s. A string of symbols in $\Sigma_3$ gives three rows of 0s and 1s. Consider each row to be a binary number and let

$$B = \{ w \in \Sigma_3^* \mid \text{the bottom row of } w \text{ is the sum of the top two rows} \}.$$ 

For example,

$$\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \in B, \quad \text{but} \quad \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \notin B.$$

Show that $B$ is a regular language.

*Hint: It is easier to think in terms of the reverse of $B$. For any language $A$, let $A^R = \{ w^R \mid w \in A \}$. You may assume that regular languages are closed under the reverse operation and think about whether $B^R$ is regular.*

**Problem 5.** We say that string $x$ is a *prefix* of string $y$ if a string $z$ exists where $xz = y$, and that $x$ is a *proper prefix* of $y$ if in addition $x \neq y$. Consider the following operation on a language $A$:

$$\text{NOEXTEND}(A) = \{ w \in A \mid w \text{ is not the proper prefix of any string in } A \}.$$ 

Show that the class of regular languages is closed under the NOEXTEND operation.

*Hint: Recall the meaning of “closed under”; the question is asking you to show that if $A$ is a regular language, then $\text{NOEXTEND}(A)$ is also a regular language.*