In the problems below where you are asked to design an automata, you only need to give a state diagram. Further, anything marked Extra Practice is for your own benefit and will not be marked.

**Edit.** Important change to number 4 below. We should say \( x \equiv_L y \) if \( x \) and \( y \) are indistinguishable by \( L \) (an earlier version uploaded incorrectly said that \( x \equiv_L y \) if they were distinguishable by \( L \)).

1. Design a DFA\(^1\) for the following language over the alphabet \( \{a, b\} \):

\[ \{x \mid x \text{ contains an equal number of occurrences of } ab \text{ and } ba \} \]

So, for example, the string \( aba \in D \) since it has one occurrence each of \( ab \) and \( ba \); but the string \( baba \not\in D \) since it has two occurrences of \( ba \) but only one occurrence of \( ab \).

**Solution.**

---

\(^1\)You just need to submit the state diagram.
2. Suppose that both \( A, B \subseteq \{0, 1\}^* \) are regular languages. Show that the language

\[
A \setminus B = \{ x \mid x \in A, x \notin B \}
\]

is regular.

**Solution.** First, we claim that the set \( \{0, 1\}^* \setminus B \) is regular. To see this, let \( M \) be any DFA for \( B \), with set of states \( Q \) and set of accepting states \( F \). Let \( M' \) be the same DFA as \( M \), except replace \( F \) with \( Q \setminus F \) — so, accepting states of \( M' \) are the regular states of \( M \), and the regular states of \( M' \) are the accepting states of \( M \). Clearly a string \( x \) is accepted by \( M' \) if and only if it is not accepted by \( M \); thus, the new DFA \( M' \) accepts \( \{0, 1\}^* \setminus B \).

Now, observe that \( A \setminus B = A \cap (\{0, 1\}^* \setminus B) \). We proved in class that regular languages are closed under intersection, and we just showed that they are closed under complementation. Thus, since \( A \) and \( B \) are regular so is \( A \setminus B \).

3. Give non-deterministic finite automata for the following two languages over \( \{0, 1\} \):

(a) \( \{00\} \) (that is, the language that contains *only* the string 00).

**Solution.**

- \( q_\varepsilon \) start
- \( 0 \rightarrow q_0 \)
- \( 0 \rightarrow q_{00} \)

(b) \( \{x \mid \text{The second-last character of } x \text{ is a 0}\} \).

**Solution.**

- \( q_0 \) start
- \( 0 \rightarrow q_1 \)
- \( 1 \rightarrow q_2 \)

(Extra Practice: For the second language, also give a DFA. What can you say about the DFA vs. the NFA?)

4. Let \( \Sigma \) be an alphabet. Let \( x \) and \( y \) be strings over \( \Sigma \) and let \( L \subseteq \Sigma^* \) be a language over \( \Sigma \). We say that \( x \) and \( y \) are distinguishable by \( L \) if there is a string \( z \) such that exactly one of the strings \( xz, yz \) is in \( L \). If \( x \) and \( y \) are not distinguishable by \( L \) then we write \( x \equiv_L y \).

(a) Consider the language \( L = \{x \mid x \text{ contains the string } 010\} \) over \( \{0, 1\}^* \). Give two distinct strings \( x \neq y \) which are distinguishable by \( L \), and two distinct strings \( x' \neq y' \) which are not distinguishable by \( L \). Explain your answer.

**Solution.** The strings \( x = 010 \) and \( y = 111 \) are distinguishable by \( L \), which is easily seen by choosing \( z = \varepsilon \): clearly \( xz = x \in L \) and \( yz = y \notin L \). On the other hand, the strings \( x = 010 \) and \( y = 1010 \) are not distinguishable by \( L \). Indeed, since 010 is a substring occurring in both \( x \) and \( y \), it follows that \( xz, yz \in L \) for any string \( z \).
(b) Let $x = 000$ and $y = 111$. Give an infinite regular language $L \subseteq \{0, 1\}^*$ such that $x$ and $y$ are distinguishable by $L$, and another infinite regular language $L'$ such that $x$ and $y$ are not distinguishable by $L' \subseteq \{0, 1\}^*$. Explain your answer.

**Solution.** An infinite language $L$ which distinguishes $x$ and $y$ is $L = \{0^n \mid n \geq 0\}$: choosing $z = \varepsilon$ we have $xz = x \in L$ and $yz = y \notin L$. On the other hand, an infinite language $L$ which does not distinguish $x$ and $y$ is $\{0, 1\}^*$: for this, it is obvious that $xz, yz \in L$ for any string $z$.

(c) Show that for any language $L$, $\equiv_L$ is an equivalence relation (that is, it is reflexive, symmetric, and transitive).

**Solution.** The relation $\equiv_L$ is reflexive by definition: if $x$ and $z$ are any strings, then clearly we can not have both $xz \in L$ and $xz \notin L$. Similarly, if $x, y$ are strings such that $x \equiv_L y$, then it follows that for all $z$ we have that either $xz, yz \in L$ or $xz, yz \notin L$. From this it immediately follows that $y \equiv_L x$, and so the relation is symmetric.

Finally, to see transitivity, suppose that $a \equiv_L b$ and $b \equiv_L c$, and we show that $a \equiv_L c$. Let $z$ be any string. Since $a \equiv_L b$, it follows that either both $az, bz \in L$ or $az, bz \notin L$. If both $az$ and $bz$ are in $L$, then since $b \equiv_L c$ it follows that $cz \in L$, and thus both $az, cz \in L$. On the other hand, if $az, bz \notin L$, then again since $b \equiv_L c$ it follows that $cz \notin L$, and thus both $az, cz \notin L$. Thus $a \equiv_L c$. 

3