You have 90 minutes to complete this exam. You may state without proof any fact taught in class or assigned as homework.

1. Find a string of minimum length not in the language $1^{*}(0 \cup 10)^{*}1^{*}$.

2. Give a regular expression for each of the following languages:
   a. odd-length binary strings that begin and end with the same symbol;
   b. binary strings in which every 0 is immediately preceded by a 1 and immediately followed by a 1.
Let $L$ be the set of odd-length binary strings in which the first, middle, and last symbols are the same. Use the pumping lemma to prove that $L$ is nonregular. You must not use the Myhill–Nerode theorem or closure properties.

Design an algorithm that takes as input a DFA $D$ and determines whether $D$ accepts infinitely many strings. Your algorithm need not be efficient, but it must terminate.
Construct a DFA for $\Sigma(\Sigma \Sigma)^* \cup \Sigma^* 1$ with the smallest possible number of states, where $\Sigma = \{0, 1\}$. Prove that your DFA is the smallest possible.

True or false? Prove your answer.

(a) If $L$ is a nonregular language, then so is its reverse $L^R$.

(b) If $L$ is a nonregular language, then so is the language $L'$ of all strings obtained by taking a nonempty string in $L$ and removing a symbol from it.

(c) If $L_1 \subseteq \{a, b\}^*$ and $L_2 \subseteq \{c, d\}^*$ are nonregular languages, then so is their concatenation $L_1L_2$ over the alphabet $\{a, b, c, d\}$. 
For each of the following languages $L$ over the binary alphabet, determine whether it is regular and prove your answer:

(2 pts) **a.** strings of the form $0^n1^m$ for some integers $n, m$ with $m \neq 2018n$;

(2 pts) **b.** strings in which every zero is at distance 2 from some other zero, e.g., $\varepsilon$, 010, 0000, 000010 but not 100, 000, 0;

(2 pts) **c.** strings in which some nonempty proper prefix is equal to some nonempty proper suffix.
You have 90 minutes to complete this exam. You may state without proof any fact taught in class or assigned as homework.

1. Find a string of minimum length not in the language $1^*(0 \cup 10)^*1^*$.

   \textit{Solution.} 0110

2. Give a regular expression for each of the following languages:

   (2 pts) a. odd-length binary strings that begin and end with the same symbol;

   (2 pts) b. binary strings in which every 0 is immediately preceded by a 1 and immediately followed by a 1.

   \textit{Solution.}

   a. $\Sigma \cup 0\Sigma(\Sigma\Sigma)^*0 \cup 1\Sigma(\Sigma\Sigma)^*1$.

   b. $1^* \cup 1^+(01^+)^*$. The answer $(1^+01^+)^*$ is tempting but incorrect because it misses the strings 1 and 10101, among others.
Let $L$ be the set of odd-length binary strings in which the first, middle, and last symbols are the same. Use the pumping lemma to prove that $L$ is nonregular. You must not use the Myhill–Nerode theorem or closure properties.

**Solution.** Let $p \geq 1$ be arbitrary. Consider the string $w = 10^p10^p1 \in L$. Let $w = xyz$ be any decomposition such that $y$ is nonempty and occurs within the first $p$ symbols. Then removing $y$ results either in an even-length string or in an odd-length string with middle symbol 0 and last symbol 1. Either way, $xz \notin L$. By the pumping lemma, $L$ is nonregular.

Design an algorithm that takes as input a DFA $D$ and determines whether $D$ accepts infinitely many strings. Your algorithm need not be efficient, but it must terminate.

**Solution.** Think of the given DFA $D = (Q, \Sigma, \delta, q_0, F)$ as a directed graph, by viewing the states as vertices and the transitions as edges. For a state $q \in Q$, we say that $q$ is part of a cycle if there is a nonempty path from $q$ to itself. We claim that $D$ accepts infinitely many strings if and only if there is a path from $q_0$ to some state $q'$ that is part of a cycle, and a path from $q'$ to some state $q'' \in F$. The “if” direction is straightforward: by traversing the cycle repeatedly, one can generate infinitely many strings that $D$ accepts. For the “only if” direction, suppose that $D$ accepts infinitely many strings and in particular accepts some string $w$ of length at least $|Q|$. By the argument in the pumping lemma, $D$ must enter some state $q'$ more than once while processing $w$. Then $q'$ is part of a cycle and is reachable from $q_0$, and the end state is some accept state $q''$.

With this criterion in hand, our algorithm simply considers every pair $(q', q'') \in Q \times F$ and checks whether: (i) there is a path from $q$ to $q'$; (ii) the state $q'$ is part of a cycle; and (iii) there is a path from $q'$ to $q''$. Conditions (i)–(iii) can be checked using any graph traversal method, such as depth-first search or breadth-first search. We output “infinite” if (i)–(iii) hold for some pair $(q', q'')$, and “finite” otherwise.
Construct a DFA for $\Sigma(\Sigma\Sigma)^* \cup \Sigma^*1$ with the smallest possible number of states, where $\Sigma = \{0, 1\}$. Prove that your DFA is the smallest possible.

**Solution.** The language $L = \Sigma(\Sigma\Sigma)^* \cup \Sigma^*1$ in question consists of binary strings that have odd length or end with a 1. It is recognized by the following DFA:

![DFA Diagram]

By the Myhill–Nerode theorem, no smaller DFA exists for $L$ because each of the three strings $\varepsilon, 01$ is in a different equivalence class of $\equiv_L$. Their distinguishing suffixes are as follows:

<table>
<thead>
<tr>
<th></th>
<th>$\varepsilon$</th>
<th>0</th>
<th>01</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\varepsilon$</td>
<td>$\varepsilon$</td>
<td>$\varepsilon$</td>
<td>$\varepsilon$</td>
</tr>
<tr>
<td>0</td>
<td>$\varepsilon$</td>
<td>$\varepsilon$</td>
<td>$\varepsilon$</td>
</tr>
<tr>
<td>01</td>
<td>$\varepsilon$</td>
<td>$\varepsilon$</td>
<td>$\varepsilon$</td>
</tr>
</tbody>
</table>

6. True or false? Prove your answer.

(a) If $L$ is a nonregular language, then so is its reverse $L^R$.

(b) If $L$ is a nonregular language, then so is the language $L'$ of all strings obtained by taking a nonempty string in $L$ and removing a symbol from it.

(c) If $L_1 \subseteq \{a, b\}^*$ and $L_2 \subseteq \{c, d\}^*$ are nonregular languages, then so is their concatenation $L_1L_2$ over the alphabet $\{a, b, c, d\}$.

**Solution.**

(a) True. Passing to the contrapositive, we arrive at the statement “if $L^R$ is regular, then so is $L = (L^R)^R$. This is true by the closure of regular languages under reverse.

(b) False. Define $L = 0\Sigma^* \cup \{10^n1^n : n \geq 0\}$. Then $L' = \Sigma^*$, a regular language. However, $L$ is nonregular! Indeed, for any integers $i \neq j$, we have $10^i1^j \in L$ and $10^j1^i \notin L$. Thus, each of the strings $1, 10, 100, 1000, \ldots$ is in a different equivalence class of $\equiv_L$. Since there are infinitely many equivalence classes, $L$ is nonregular by the Myhill–Nerode theorem.

(c) True. Fix nonregular languages $L_1 \subseteq \{a, b\}^*$ and $L_2 \subseteq \{c, d\}^*$. Suppose for the sake of contradiction that $L_1L_2$ is regular, with a DFA $D$. Since $L_2$ is nonempty (being nonregular!), an NFA for $L_1$ can obtained from $D$ by replacing all occurrences of $c$ and $d$ as edge labels, with $\varepsilon$. Thus, $L_1L_2$ must in fact be nonregular.
For each of the following languages $L$ over the binary alphabet, determine whether it is regular and prove your answer:

(2 pts) **a.** strings of the form $0^n1^m$ for some integers $n, m$ with $m \neq 2018n$;

(2 pts) **b.** strings in which every zero is at distance 2 from some other zero, e.g., $\varepsilon$, 010, 0000, 000010 but not 100, 000, 0;

(2 pts) **c.** strings in which some nonempty proper prefix is equal to some nonempty proper suffix.

**Solution.**

**a.** For any integers $i \neq j$, we have $0^i1^{2018i} \notin L$ but $0^j1^{2018i} \in L$. Therefore, each of the strings $\varepsilon, 0, 00, 000, \ldots$ is in a different equivalence class of $\equiv_L$. Since there are infinitely many equivalence classes, $L$ is nonregular by the Myhill–Nerode theorem.

**b.** The complement of this language is the set of binary strings in which some zero has no zeros at distance 2 to the right or to the left of it, either because those positions are occupied by ones or because the string has ended. Such strings are captured by the regular expression $(\varepsilon \cup \Sigma \cup \Sigma^*1\Sigma)0(\varepsilon \cup \Sigma \cup \Sigma1\Sigma^*)$ and therefore form a regular language. Since regular languages are closed under complement, the original language is regular as well.

**c.** For any integers $i > j$, we have $10^i1^{10j} \in L$ but $10^j1^{10j} \notin L$. Therefore, each of the strings 1, 10, 100, 1000, $\ldots$ is in a different equivalence class of $\equiv_L$. Since there are infinitely many equivalence classes, $L$ is nonregular by the Myhill–Nerode theorem.