Problem 1

Answer. Consider the language \( L = \{1^{k-2}\} \).

First we give a DFA \( M = (Q, \Sigma, \delta, q_0, F) \) that recognizes \( L \), where \(|Q| = k\).

Let \( Q = \{q_0, q_1, \ldots, q_{k-1}\} \) and \( \Sigma = \{0, 1\} \).

The transition function \( \delta \) is defined as follows. Let \( \delta(q_i, 1) = q_{i+1} \) and \( \delta(q_i, 0) = q_{k-1} \), for \( i = 0, 1, \ldots, k-2 \). Let \( \delta(q_{k-1}, 0) = q_{k-1}, \delta(q_{k-1}, 1) = q_{k-1} \).

Let \( q_0 \) be the start state.

Let \( F = \{q_{k-2}\} \).

There is only one path from the start state to the accept state. The path is (\( q_0, q_1, \ldots, q_{k-2} \)). So \( M \) can accept and only accept \( 1^{k-2} \). This shows that \( M \) recognizes \( L \).

On the other hand, assume we have another DFA \( M' = (Q', \Sigma, \delta', q'_0, F') \) which recognizes \( L \) but only has \( k-1 \) states. Let the sequence of states (path \( P \)), following \( M' \) reading \( 1^{k-2} \), be \( p_0 = q'_0, p_1, \ldots, p_{k-2} \) where \( p_{k-2} \in F' \). Consider \( q' = \delta(q'_0, 0) \). We must have \( q' \notin \{p_0, p_1, \ldots, p_{k-2}\} \). Otherwise \( M' \) will accept another string which starts with a 0. Thus \( p_0, p_1, \ldots, p_{k-2} \in Q - \{q'\} \). As \( |Q - \{q'\}| = k - 2 \), at least two of \( p_0, p_1, \ldots, p_{k-2} \) are equal. This shows that there is a cycle in \( P \). So \( M' \) accepts more than one string, contradicting that \( M' \) recognizes \( L \).

 Remark 1. If one’s counterexample does not work, one can get at most 5 points.

One has to prove the counter example is a language that can be recognized by a DFA of \( k \) states and cannot be recognized by any DFA with \( k-1 \) states. Each part of the proof is worth 5 points.

Problem 2

Answer. (a) Assume this language is \( L \) and it is regular. According to the pumping lemma, there exists a constant \( p \) such that \( \forall s \in L, |s| \geq p, s \) can be broke into three sub-strings \( s = xyz \) such that

- \(|y| > 0\)
- \(|xy| \leq p\)
- \(\forall i \geq 0, xy^iz \in L\)

Let \( s = 0^p10^p \). So \( y = 0^l, 1 \leq l \leq p \). Consider \( xy^2z \). It should be in \( L \) according to the pumping lemma. However \( xy^2z = 0^{p+1}10^p, n_1 \neq p \). This is a contradiction. So \( L \) is not regular.

(b) Assume this language is \( L \) and it is regular. Let the pumping length of \( L \) be \( p \).

Consider the string \( s = 0^p10^{p+p} \). As \( p \neq p + p \), \( s \) is not a palindrome. So \( s \) is in \( L \). By the pumping lemma, \( s \) can be broke into \( xyz \) where \( |xy| \leq p, |y| > 0 \). We know \( y = 0^i \) where \( i \) is an integer in \([1, p]\). Also by the pumping lemma, \( xy^i0^iz \) is in \( L \). However \( xy^i0^i0^iz = 0^{p+p}10^{p+p} \) which is a palindrome. This is a contradiction.

As a result, \( L \) is not regular.

Remark 2. If one misunderstands the pumping lemma, one can get at most 5 points.

Each question is worth 10 points. For each question, if one’s proof does not use the pumping lemma correctly, one can get at most 3 points for that question.
Problem 3

Answer. (a) Yes. We show that $B = L$ where $L = 10^1(0|1)^*$. For every $s \in B$, let $s = 1^k y$. If $k = 1$, we know $y$ has at least one 1. So $y \in 0^*1(0|1)^*$. Thus $s \in L$. If $k \geq 2$, $1^k$ is in $10^*1(0|1)^*$ and $y$ is in $(0|1)^*$. So $s = 1^k y \in L$. On the other hand, for any $s \in L$, as $s$ starts with a 1, $s$ can be viewed as $1y$ where $y \in 0^*1(0|1)^*$. Thus $y$ contains at least one 1. According to the definition of $B$, $s \in B$. So $L = B$.

(b) No. We use the pumping lemma. Let $p$ be the pumping length of $C$. Let $s = 1^p 01^p \in C$. By the pumping lemma, $s = xyz$ where $|xy| \leq p$. So $y = 1^l$ for some $l \in [1, p]$. By the pumping lemma, $xz \in C$. However, $xz$ does not have the form $1^{k'}a'$ with $a'$ has at most $k'$ 1s. So $C$ is not regular.

Remark 3. Each question is worth 10 points.

For each question, if one makes a wrong judgement, one loses all the points of that question.

For question (a), if one gives a wrong DFA/NFA or regular expression, one can get at most 3 points.

For question (b), if one does not correctly use the pumping lemma, one can get at most 3 points.

Problem 4

Answer. (a) The grammar is as follows.

- $S \rightarrow 0A0|1A1|0|1$
- $A \rightarrow A0|A1|\epsilon$

(b) The grammar is as follows.

- $S \rightarrow 0S0|1S1|0|1|\epsilon$

Remark 4. Each question is worth 10 points.

For each question, if one misunderstands the definition of grammar for context free language, or one’s grammar does not work generally, one can get at most 3 points.

In both questions, if one’s grammar includes/excludes the empty string, please do not take off points.

Other grammars may also work, please grade accordingly.

Problem 5

Answer. The grammar is as follows.

- $S \rightarrow 0S0|1S1|A$
- $A \rightarrow 0B1|1B0$
- $B \rightarrow 0B0|0B1|1B0|1B1|\epsilon$
First we prove that every string \( w = w_0w_1 \cdots w_{n-1} \) generated by the grammar is in \( D \).

Notice that for every rule in the grammar, the number of terminals in the right hand side is even. As a result, \( w \) has even length. Assume \( w = xy \). In the procedure of generating \( w \), the rule \( A \rightarrow 0B1|1B0 \) has to be applied for at least once. This means that there exists an index \( i < n/2 \), such that \( w_i \neq w_{n-1-i} \). Thus \( x \neq y^R \). So \( w \in D \).

Next we prove that every string \( w \) in \( D \) can be generated by the grammar.

According to the definition of \( D \), there exists an index \( i < n/2 \), such that \( w_i \neq w_{n-1-i} \) and \( \forall j \leq i-1, \ w_j = w_{n-1-j} \). We give a process generating \( w \) following the rules in our grammar. We generate the first \( i-1 \) characters and the last \( i-1 \) characters of \( w \) following \( S \rightarrow 0S0|1S1 \). This can be done because by following \( S \rightarrow 0S0|1S1 \), for every string \( u \), we can get \( uSu^R \). Then we apply the rule \( S \rightarrow A \) and \( A \rightarrow 0B1|1B0 \) to generate \( w_i \) and \( w_{n-1-i} \). After that we generate the remaining bits of \( w \) by following \( B \rightarrow 0B0|0B1|1B0|1B1|\epsilon \). This can be done because any even length string can be generated by following \( B \rightarrow 0B0|0B1|1B0|1B1|\epsilon \). As a result, \( w \) can be generated by our grammar.

This shows that \( D \) is a context-free language.

\[ \square \]

**Remark 5.** If one’s grammar does not work, one can get at most 5 points. The key point here is that one’s grammar should reflect the property that \( x \) and \( y^R \) are different in at least one position.

One needs to prove that any string generated by the grammar is in \( D \) and any string in \( D \) can be generated by the grammar. For each direction, if one’s proof is wrong or missing, please take off 5 points.

One can also use PDA for this problem, please grade accordingly.