1. Design a DFA for the language \( L = \{ w \in 0^* \mid |w| \text{ is a multiple of } 2 \text{ or } 3 \} \). Briefly explain the construction. (5)

One clean and intuitive construction will be to take product construction of the \( M_1 \) and \( M_2 \) that recognize multiple of 2 and 3 respectively and mark out the final states as either final state of \( M_1 \) or \( M_2 \).

2. Let \( L_1, L_2 \subset \Sigma^* \) be infinite languages. (5+5)

   (a) If \( L_1 \cap L_2 \) is regular then \( L_1 \) and \( L_2 \) are regular. Justify or give a counterexample.
   False. Consider \( L_1 = 0^11^1 \) (some non regular language) and \( L_2 = \Sigma^* \). Then \( L_1 \cup L_2 = \Sigma^* \Rightarrow L_1 \cap L_2 = \Sigma^* = \phi \). Note that \( L_1 \) is not regular and \( \phi \) is regular.

   (b) For \( L_1 \subset L_2 \), can \( L_2 \) be non-regular and \( L_1 \) regular ? Provide an example or argue about impossibility.
   Let \( L_2 = 0^p \) where \( p \) is prime and \( p \neq 2 \cup L_1 = (00)^* \). Note that \( L_2 \) is not regular, otherwise \( L_2 - L_1 \) is regular.
3. Consider the language \( L = \{0^i \cdot 1^j | i \neq j\} \) for \( \Sigma = \{0, 1\} \). Consider the following arguments to show that \( L \) is not regular. Point out the fallacy in the proofs (if any) in one sentence.

(a) Since \( \{0^i \cdot 1^j | i = j\} \) is not regular (proved in class), it follows from the the closure property of complement of Regular languages that \( L \) is not regular. \( \boxed{2} \)

\( \{0^i \cdot 1^j | i \neq j\} \) is not the complement of \( \{0^i \cdot 1^j | i = j\} \) for the alphabet \( \{0, 1\} \). In particular it doesn’t contain strings not of the form \( 0^*1^* \).

(b) Consider the language \( L_< = \{0^i \cdot 1^j | i < j\} \). It can be proved easily by Pumping Lemma that \( L_< \) is not regular by choosing a string \( 0^n \cdot 1^{n+1} \) where \( n \) is the constant of the Pumping Lemma and pumping enough 0’s so that it exceeds the number of 1’s. Similarly, the language \( L_> \{0^i \cdot 1^j | i > j\} \) is not regular. Since \( L = L_< \cup L_> \), it follows that \( L \) is not regular. \( \boxed{2} \)

Even if \( L_1, L_2 \) are not regular, their union can be regular - for example a non regular language and its complement.

(c) Using Pumping Lemma Consider a string \( z = 0^n \cdot 1^{n+k} \) where \( n \) is the constant of the Pumping Lemma and \( k \) is an integer \( 1 \leq k \leq n \). In the partition \( z = u \cdot v \cdot w \), note that \( uv \) consists only of 0’s, so choose \( v = 0^k \). Then \( uv^2w = 0^{n+k} \cdot 1^{n+k} \notin L \) and therefore a contradiction. \( \boxed{2} \)

We can’t choose \( v \) in a pumping lemma proof and must argue wrt all possible \( v \)s.

(d) In case, you find all the proofs are incorrect, then either show that \( L \) is regular or give a correct proof that \( L \) is not regular. (Otherwise you just mention one of the previous proof that is correct). \( \boxed{6} \)

Since \( 0^*1^* \) is regular, its complement is also regular - denote it by \( S \). Suppose \( L = \{0^i \cdot 1^j | i \neq j\} \) is regular, then \( S' = S \cup L \) is also regular using the closure properties of regular languages. Since \( \Sigma^* = S' \cup \{0^i \cdot 1^i\} \), it implies that \( \{0^i \cdot 1^i\} = \Sigma^* - S' \) is regular which is a contradiction.
4. Let $L$ be a regular language over $\{0, 1\}$ and consider the set of strings $S = \{y | y \cdot (01^*01 + 010^*) \in L\}$.

(i) What can you say about $S$ - is it always regular? Justify or give a counterexample. (10)

Consider a DFA $M$ such that $L(M) = L$. Let $R$ denote the set of strings $(01^*01 + 010^*)$ For any state $q \in Q$, if there is a string $y \in R$ such that $\delta(q, y) \in F$, then mark that state $q$ as a final state - denote this DFA by $M'$. So $M'$ is identical to $M$ except perhaps the set of final states $F'$.

**Proof of correctness** If a string $y$ is accepted by this machine then from our definition of final states $F'$ there must be a string $x \in R$ such that $\delta(q_0, y \cdot x) \in F$, i.e., $y \cdot x \in L$.

Conversely, if for any string $y$, there exists $x \in R$ such that $\delta(q_0, y \cdot x) \in F$ (i.e. it belongs to $L$) then $\delta(q_0, y) \in F'$ and is accepted by $M'$.

(ii) What can you say about $S' = \{y | y \cdot 0^i \cdot 1^i \in L, i \geq 1\}$ (3)

In the previous part, we didn’t use any property of $L$ being regular, so it carries over this case also.

However, in part 1, we can procedurally determine if there is a path between $q$ and $F$ with one of the strings in $L$. Consider a state $p \in Q$ - we can find an r.e. for all the paths from $q$ to $F$, say $R'$. Then, if $R \cap R' \neq \phi$, then $q \in F'$.

For the second problem, $R$ is not an r.e. and we may not know how to find $R \cap R'$, nevertheless, the definition of $F'$ is still valid and the machine is a DFA by construction. This can be thought of as a non-constructive proof.