(a) For the given languages $L_1 = \emptyset$, $L_2 = \varepsilon$, and $L_3 = \{0, 1\}^*$. Compute $L_1, L_2$ and $L_2 \cup L_3$.

(b) Construct a DFA for the language that contains the strings ending with 0.

(c) Define the language of the following finite automaton.

\[
\begin{array}{c}
\text{0, 1} \\
\text{0, 1} \\
\end{array}
\]

\[
\begin{array}{c}
q_0 \\
q_2 \\
q_3 \\
\end{array}
\]

(d) Let $M = (Q, \Sigma, q_0, F, \delta)$ be an NFA. Show that for any $q \in Q$ and $a \in \Sigma$,

\[\hat{\delta}(q, a) = \delta(q, a)\]

(e) From the given NFAs, decide whether the two accept the same language.
(f) Let \( L = \{0, 1\} \{0, 1\}^* \); construct NFA with \( \varepsilon \) moves that accepts \( L^2 \).

2. Attempt any two parts of the following:

(a) Construct a DFA accepting the following language:

\[(010 + 00)^*(10)^*\]

(b) Let \( r_1 \) and \( r_2 \) be two regular expressions defined as follows:

\[ r_1 = (00*1)*1 \]

and \( r_2 = 1 + 0(0 + 10)^*11 \).

Prove that \( r_1 = r_2 \).

(c) Prove that the language

\[ L = \{0^n \mid n \text{ is prime} \} \]

is not regular.

3. Attempt any two parts of the following:

(a) Find a Context Free Grammar (CFG) generating the following language:

\[ L = \{a^ib^ic^k \mid i = j \text{ or } i = k \} \]

(b) Describe the language generated by the following grammar:
S → bS/аА/ε
A → aA/bB/b
B → bS

(c) Show that the given grammar is ambiguous. Also find an equivalent unambiguous grammar.

S → ABA
A → aA/ε
B → bB/ε

4. Attempt any two parts of the following:

(a) Define a Push Down Automaton (PDA). Construct a PDA accepting the language of palindromes.

(b) Construct a deterministic PDA for the following language:

\[ L = \{ x \in \{a, b\}^* \mid n_a(x) \neq n_b(x) \} \]

where \( n_a(x) \): number of a's in the string x

\( n_b(x) \): number of b's in the string x

(c) Show that if L is a language of Deterministic PDA (DPDA) and R is regular then \( L \cap R \) is a language of DPDA.

5. Attempt any two parts of the following:

(a) Construct a turing machine for reversing a string.

(b) Let \( T_1 \) and \( T_2 \) be two Turing machines; compute the functions \( f_1 \) and \( f_2 \) from N to N (where N is a natural number), respectively, construct a Turing machine that computes the function \( \min(f_1, f_2) \).

(c) Prove that every recursively enumerable language where complement is closed must be recursive.