1. Attempt all question parts: \(10 \times 2 = 20\)

(a) Draw the model diagram for finite automata and pushdown automata.

(b) What is the role of finite automata for searching a keyword in documents?

(c) Write Regular Expression for the following languages:

(i) Set of all strings such that the number of 0's is odd

(ii) Set of all strings that do not contain 1101 as a substring

(d) Design CFG for the language consisting of all strings of even length over \(\{a, b\}\).

(e) Briefly write about Church-Turing thesis with a neat diagram.

(f) What is Moore and Mealy machine?

(g) Convert the given CFG into PDA by empty stack:

\[
G : \quad S \rightarrow AB \mid a \\
A \rightarrow SaS \mid \epsilon \\
B \rightarrow b
\]

(h) Define the languages generated by Turing Machine.

(i) State whether the following instances of PCP has a solution. Justify.

Top = (10, 011, 101) \quad Bottom = (101, 11, 011)

(j) Show that \(\emptyset^*\) is \(\emptyset\) by constructing its NFA using Thomson's method.
SECTION – B

2. Attempt any three question parts:

(a) (i) Explain in detail about various models of Turing Machine.
(ii) State halting problem. Prove that “HALT\text{TM} is undecidable”.

(b) Convert the following given Non-deterministic finite automata into minimized Deterministic finite automata.

<table>
<thead>
<tr>
<th>States/Input</th>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>\rightarrow p</td>
<td>{q, s}</td>
<td>{q}</td>
</tr>
<tr>
<td>*q</td>
<td>{r}</td>
<td>{q, r}</td>
</tr>
<tr>
<td>r</td>
<td>{s}</td>
<td>{p}</td>
</tr>
<tr>
<td>*s</td>
<td>-</td>
<td>{p}</td>
</tr>
</tbody>
</table>

(c) Define Inference, Derivation and Syntax tree. Consider the context free grammar
G : P \rightarrow 0P0 | 1P1 | \epsilon and build derivation, syntax tree and inference for the string 0110 using grammar G.

(d) Construct Pushdown automata for the language \( L = \{ \text{ww}^R \mid w \text{ is in } (0 + 1)^* \} \). Give instantaneous description of the input 1111.

(e) Consider the following DFA and find its equivalent regular expression using Rij method.

Given DFA:

![DFA Diagram]

SECTION – C

Attempt all questions:

3. Attempt any two parts:

(a) Draw NFA- C transition diagram for the following regular expressions:

(i) \(((ab)^* / b)a\)
(ii) \((0^* / 1^*)^* 10\)

(b) Write an algorithm to minimize the given DFA using subset construction method.

(c) Construct DFA that accepts set of natural numbers which are divisible by 3.
4. Attempt any one part:  

(a) State pumping lemma. Prove that the language consisting of “set of all strings over \{a\} whose length is prime” is not regular.  
(b) Illustrate in detail about all the closure properties of regular languages.

5. Attempt any one part:  

(a) Write down the steps required to convert CFG into Chomsky Normal Form. Consider the following CFG and find its equivalent CNF and GNF.

\[ G : \quad S \rightarrow ASB \mid \epsilon \]

\[ A \rightarrow aAS \mid a \]

\[ B \rightarrow SbS \mid A \mid bb \]

(b) How to eliminate useless symbols and unit productions in a grammar?  
Eliminate unit productions for the following CFG:

\[ G : \quad E \rightarrow E + T \mid T \]

\[ T \rightarrow T \ast F \mid F \]

\[ F \rightarrow (E) \mid I \]

\[ I \rightarrow a \mid b \mid Ia \mid Ib \mid I0 \mid I1 \]

6. Attempt any one part:  

(a) Prove that “Language L has a PDA that accepts it by final state if L has a PDA that accepts it by empty stack”.
(b) Find equivalent CFG of the following given PDA:

PDA  \( P = ( \{q_0, q_1\}, \{a, b\}, \{a, Z_0\}, \delta, q_0, Z_0) \)

Where \( \delta \):

\[ \delta (q_0, a, Z_0) = (q_0, aZ_0) \]

\[ \delta (q_0, a, a) = (q_1, aa) \]

\[ \delta (q_1, a, a) = (q_1, \epsilon) \]

\[ \delta (q_1, \epsilon, Z_0) = (q_1, \epsilon) \]
7. Attempt any two parts:  

(a) State Post Correspondence Problem and prove that "PCP is undecidable".

(b) Design transition diagram for the language \( L = \{a^i b^j c^k \mid i, j > 0 \text{ and } k = i^* j \} \) using Turing Machine.

(c) Prove that "Every language defined by a regular expression is also defined by a finite automation".