57625

Theory of Computer Science

1. a) Chomsky Hierarchy
   - Type
   - Languages
   - Form of productions in grammar
   - Accepting device

b) Give five differences between NFA and DFA.

c) Definition of R.E.

R. E.

$$a\cdot(a+b+c)^{*}\cdot(b+c)+b\cdot(a+b+c)^{*}\cdot(a+c)+c\cdot(a+b+c)^{*}\cdot(a+b)$$

d) Post correspondence problem (PCP)

PCP is an undecidable decision problem, where making decision becomes difficult.

Assume two strings over some alphabets.

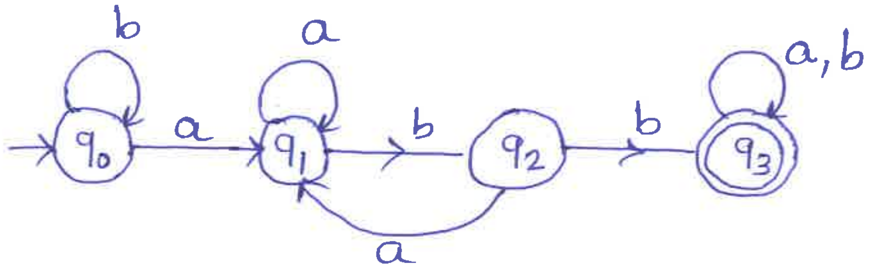
$W=\{w_1, w_2, \dots w_n\}$ $V=\{v_1, v_2 \dots v_n\}$

Problem is to find concatenation of one or more strings from the set W in any order where the corresponding concatenation is same for strings of V.

$w_1 \cdot w_3 \cdot w_5$ & $v_1 \cdot v_2 \cdot v_3$ then will they be same? Not always.

Q.2 (a) Design FA for $(a+b)^* abb (a+b)^*$

	a	b
q_0	q_1	q_0
q_1	q_1	q_2
q_2	q_1	q_3
q_3	q_3	q_3



b) Let L be a regular language, then there exists a constant of pumping lemma 'n' such that for any sufficiently longer string z in L there exists a loop that can be pumped for any number to produce strings in the RL

Let L be Regular lang.
n is pumping constant
z be sufficiently longer string in L.

(1) $z = uvw$

(2) for all $uv^iw \in L$, $|z| \geq n$ $n \geq |v| \geq 1$ $n \geq |uv|$

$z = aabbb$

$u = a$ $v = a$ $w = bbb$

$i=0$ $uv^0w = abbb$

$i=2$ $uv^2w = aaabbb$

$i=3$ $uv^3w = aaaabbb \notin L$.

Hence L is not R.L.

Q.3 (a)

$S \rightarrow 0S0 / 0A / 0 / 1B / 1$

$A \rightarrow 0A / 0$

$B \rightarrow 1B / 1$

Test whether 001100, 001010 are in the L(G)

For string "001100"

$S \Rightarrow 0S0$

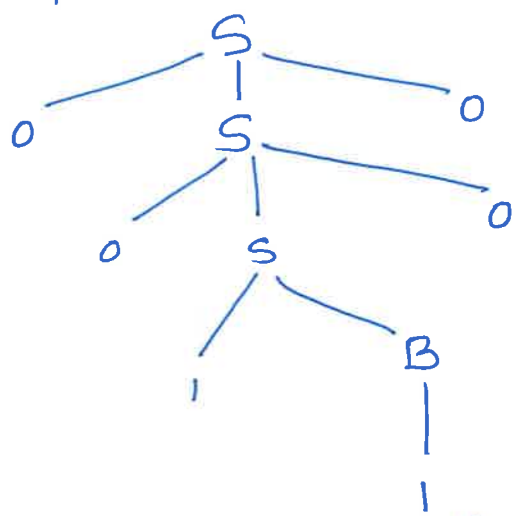
$\Rightarrow 00S00$

$\Rightarrow 001B00$

$\Rightarrow 001100$

string is generated by given grammar.

Parse Tree is

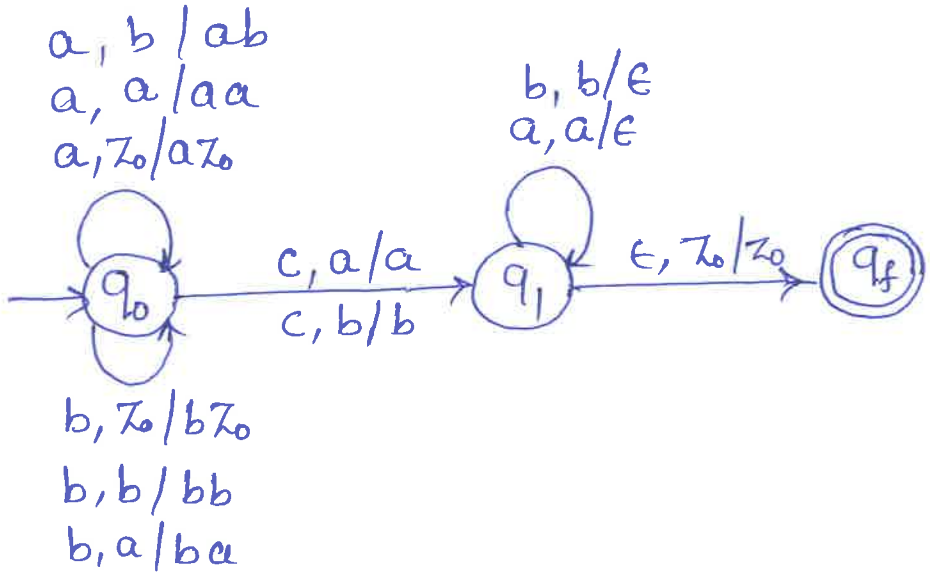


For string "001010"

$S \Rightarrow 0S0$

string cannot be generated by this grammar hence no parse tree.

Q.3(b) PDA for $L = \{ wcw^R / w \in (a,b) \text{ \& } w^R \text{ is reverse of } w \}$



Transition function

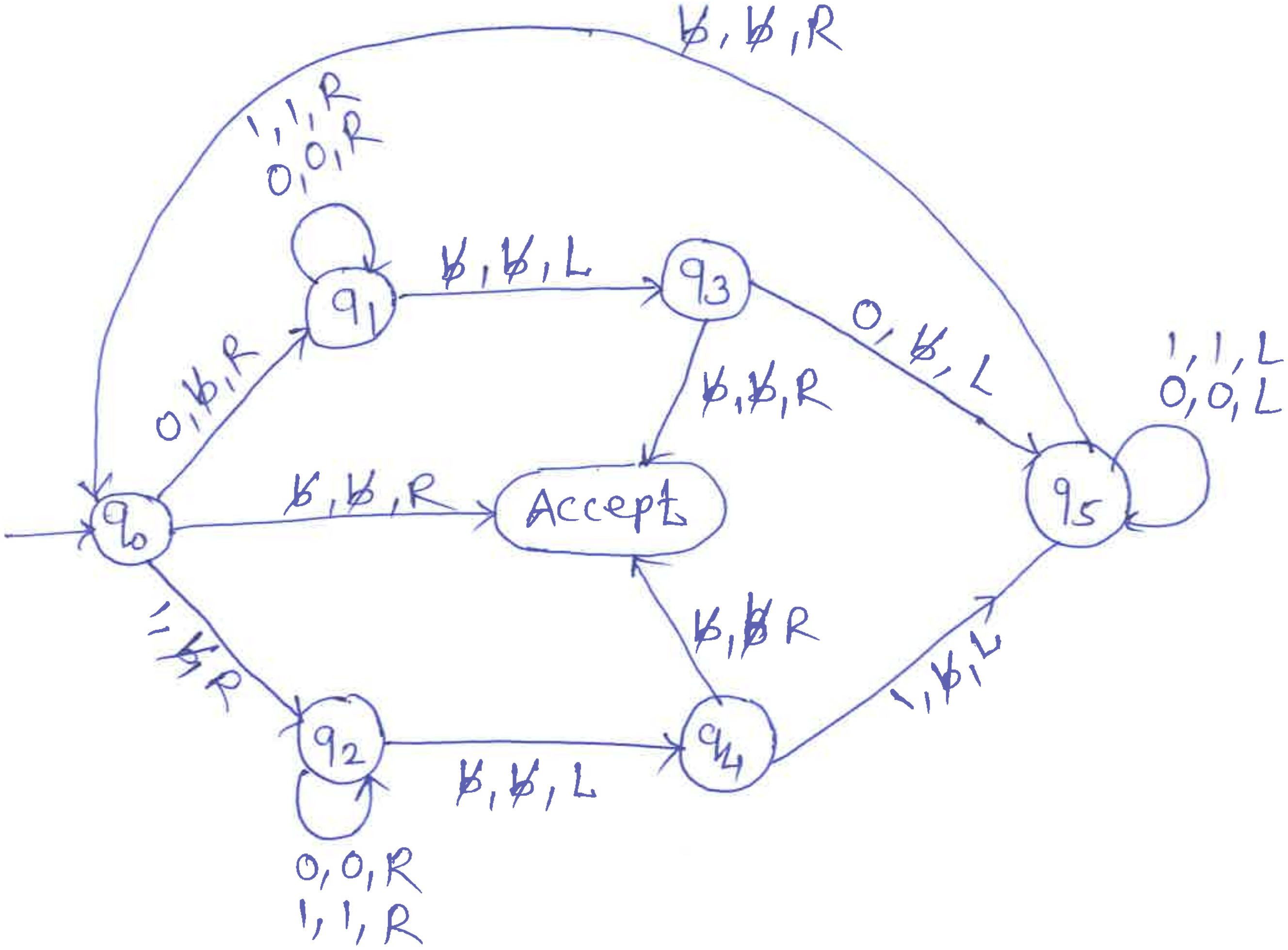
$$\delta(q_0, a, Z_0) = (q_0, aZ_0)$$
$$\delta(q_0, a, a) = (q_0, aa)$$
$$\delta(q_0, b, Z_0) = (q_0, bZ_0)$$
$$\delta(q_0, b, b) = (q_0, bb)$$
$$\delta(q_0, a, b) = (q_0, ab)$$
$$\delta(q_0, b, a) = (q_0, ba)$$

$$\delta(q_0, c, a) = (q_1, a)$$
$$\delta(q_0, c, b) = (q_1, b)$$

$$\delta(q_1, b, b) = (q_1, \epsilon)$$
$$\delta(q_1, a, a) = (q_1, \epsilon)$$

$$\delta(q_0, \epsilon, Z_0) = (q_f, Z_0)$$

Q.4 (a) TM to check palindrom over $\Sigma = \{0,1\}$



Q.4 b)

Example: Convert the following grammar G into Greibach Normal Form (GNF).

$$\boxed{\begin{array}{l} S \rightarrow XA|BB \\ B \rightarrow b|SB \\ X \rightarrow b \\ A \rightarrow a \end{array}}$$

To write the above grammar G into GNF, we shall follow the following steps:

1. Rewrite G in Chomsky Normal Form (CNF)

   It is already in CNF.

2. Re-label the variables

   S with A_1

   X with A_2

   A with A_3

   B with A_4

   After re-labeling the grammar looks like:

   $A_1 \rightarrow A_2A_3|A_4A_4$

   $A_4 \rightarrow b|A_1A_4$

   $A_2 \rightarrow b$

   $A_3 \rightarrow a$

3. Identify all productions which do not conform to any of the types listed below:

   $A_i \rightarrow A_jx_k$ such that $j > i$

   $Z_i \rightarrow A_jx_k$ such that $j \leq n$

   $A_i \rightarrow ax_k$ such that $x_k \in V^*$ and $a \in T$

4. $A_4 \rightarrow A_1A_4$ .... identified

5. $A_4 \rightarrow A_1A_4|b$.

   To eliminate A_1 we will use the substitution rule $A_1 \rightarrow A_2A_3|A_4A_4$.
   Therefore, we have $A_4 \rightarrow A_2A_3A_4|A_4A_4A_4|b$
   The above two productions still do not conform to any of the types in step 3.

   Substituting for $A_2 \rightarrow b$

   $A_4 \rightarrow bA_3A_4|A_4A_4A_4|b$

   Now we have to remove left recursive production $A_4 \rightarrow A_4A_4A_4$

$A_4 \rightarrow bA_3A_4|b|bA_3A_4Z|bZ$

$Z \rightarrow A_4A_4|A_4A_4Z$

6. At this stage our grammar now looks like

   $A_1 \rightarrow A_2A_3|A_4A_4$

   $A_4 \rightarrow bA_3A_4|b|bA_3A_4Z|bZ$

   $Z \rightarrow A_4A_4|A_4A_4Z$

   $A_2 \rightarrow b$

   $A_3 \rightarrow a$

   All rules now conform to one of the types in step 3.
   But the grammar is still not in Greibach Normal Form!

7. All productions for A_2, A_3 and A_4 are in GNF

   <u>for $A_1 \rightarrow A_2A_3|A_4A_4$</u>

   Substitute for A_2 and A_4 to convert it to GNF

   $A_1 \rightarrow bA_3|bA_3A_4A_4|bA_4|bA_3A_4ZA_4|bZA_4$

   <u>for $Z \rightarrow A_4A_4|A_4A_4Z$</u>

   Substitute for A_4 to convert it to GNF

   $Z \rightarrow bA_3A_4A_4|bA_4|bA_3A_4ZA_4|bZA_4|bA_3A_4A_4Z|bA_4Z|bA_3A_4ZA_4Z|bZA_4Z$

8. Finally the grammar in GNF is

   $A_1 \rightarrow bA_3|bA_3A_4A_4|bA_4|bA_3A_4ZA_4|bZA_4$

   $A_4 \rightarrow bA_3A_4|b|bA_3A_4Z|bZ$

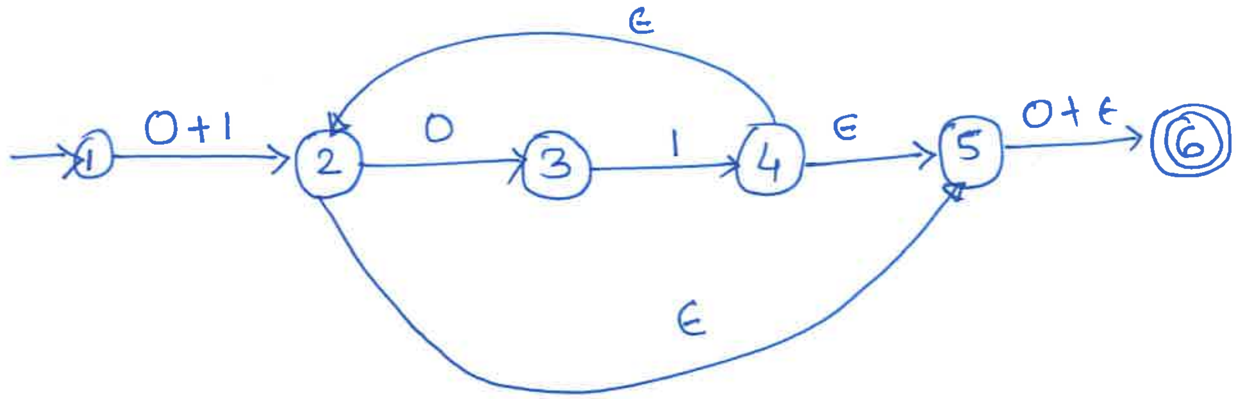
   $Z \rightarrow bA_3A_4A_4|bA_4|bA_3A_4ZA_4|bZA_4|bA_3A_4A_4Z|bA_4Z|bA_3A_4ZA_4Z|bZA_4Z$

   $A_2 \rightarrow b$

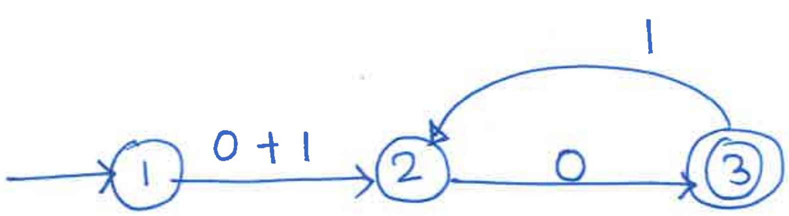
   $A_3 \rightarrow a$

Q. 5 (a) NFA with ε-moves and DFA.

$(0+1)(01)^*(0+\epsilon)$



Minimized DFA will be



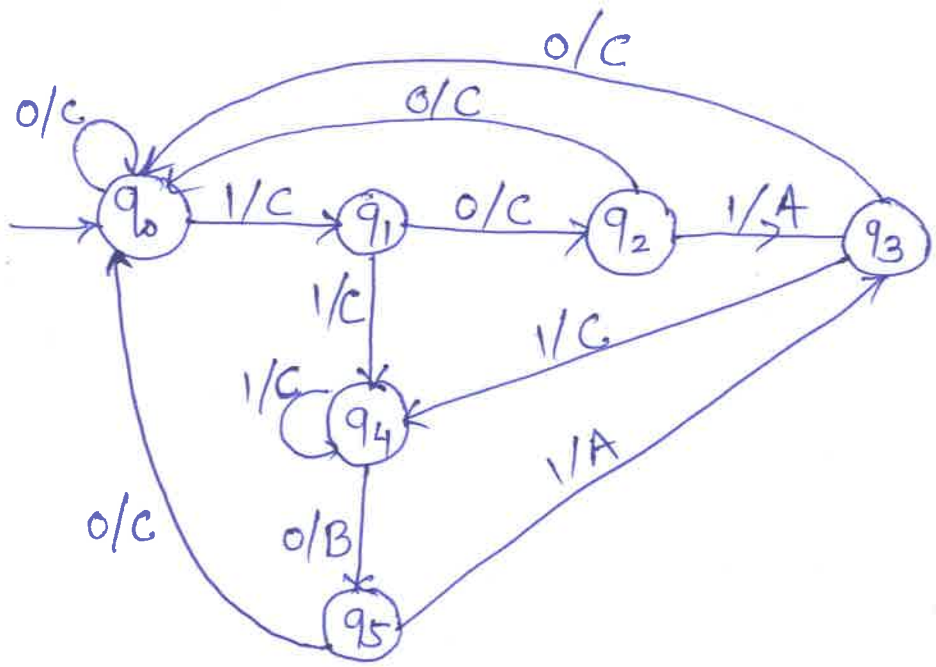
	0	1
1	2	2
2	3	–
* 3	–	2

Q.5(b) Mealy m/c over $\Sigma = \{0, 1\}$

Output: A if input ends in 101

B if input ends in 110

C otherwise



Q.6 a) closure properties of CFL.

Explain closure properties of CFL like Concatenation, union etc.

b) Myhill Nerode Th^m

Explain with example

c) Rice's Th^m

Explain with example

d) Moore & Mealy M/C.

Define & differentiate between Moore & Mealy

e) Variant of T.M.

Explain different Turing Machines.