Fifth Semester B.E. Degree Examination, CBCS - Dec 2017 / Jan 2018

Automata Theory & Compatibility

Time: 3 hrs. Max. Marks: 80

Note : *Answer any FIVE full questions, selecting ONE full question from each module.*

Module - 1

1\. a. **Define the following terms with examples:**
(i) Alphabet (ii) Power of an alphabet
(iii) Concatenation (iv) Languages **(04 Marks)**

Ans. **i. Alphabet :** A language consists of various symbol from which the words, statements etc. can be obtained. These symbols are called Alphabets. The symbol Σ denotes the set of alphabets of a language.

Ex : Σ = {a,b,......z, A.B;.......,Z,0.....9,#,C,),}....etc}

ii Power of an alphabet : If Σ is an alphabet, we can epress the set of all strings of a certain length from that alphabet by using the exponential notation.

Ex: Σ = {0,1} the

Σ^1 = {0,1}, Σ^2 = {00,01,10,11}

iii. Concatenation : The concatenation of two strings u and v, is the string obtained by writing the letters of string u followed by the letters of string v.

$u = a_1a_2a_3.....a_n \quad v = b_1b_2b_3.....b_n$

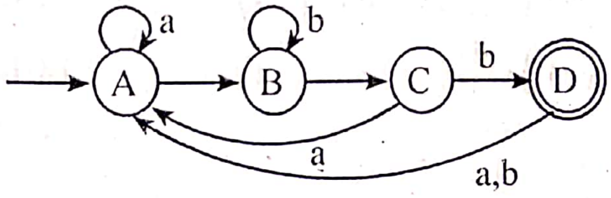
$uv = a_1a_2a_3......a_nb_1b_2b_3.....b_n$

iv. Language : A language can be defined as a set of strings obtained from Σ^* where Σ is set of alphabets of a particular language.

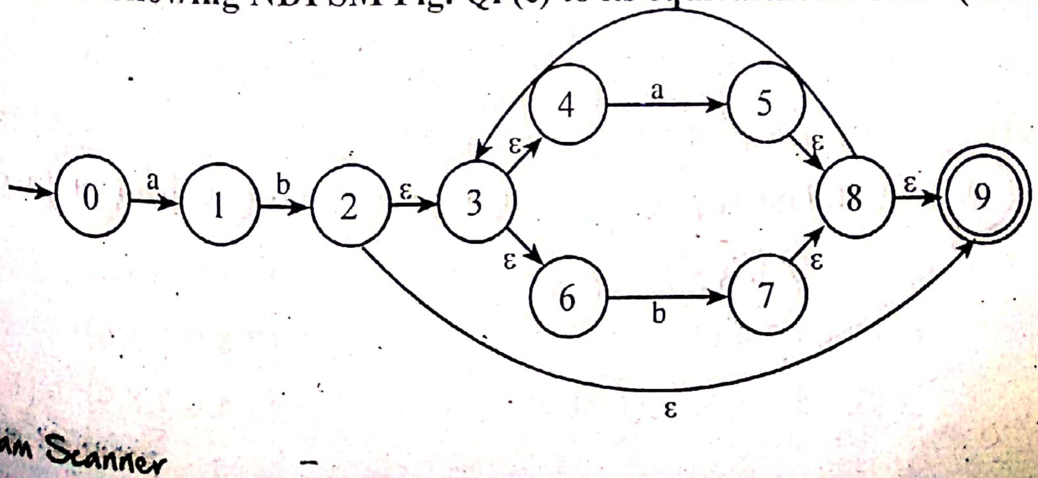
Ex : {ε,01,10,0011,1010,0101,0011,.....}

b. **Draw a DFA to accept strings of a's and b's ending with 'bab'.** **(03 Marks)**

Ans.



c. **Convert the following NDFSM Fig. Ql (c) to its equivalent DFSM.** **(09 Marks)**



Ans. Consider the state A :

When input is a :

$$\begin{aligned}\delta(A,a) &= \text{ECLOSE}(\delta_E(A,a))\\ &= \text{ECLOSE}(\delta_E(0,a))\\ &= \{1\} \rightarrow (B)\end{aligned}$$

When input is b :

$$\begin{aligned}\delta(A,b) &= \text{ECLOSE}(\delta_E(A,b))\\ &= \text{ECLOSe}(\delta_E(0,b))\\ &= \{\phi\}\end{aligned}$$

Consider the state B:

When input is a:

$$\begin{aligned}\delta(B,a) &= \text{ECLOSE}(\delta_E(B,a))\\ &= \text{ECLOSE}(\delta_E(1,a))\\ &= \phi\end{aligned}$$

When input is b

$$\begin{aligned}\delta(B,b) &= \text{ECLOSE}(\delta_E(B,b))\\ &= \text{ECLOSE}(\delta_E(1,b))\\ &= \text{ECLOSE}(\{2\})\\ &= \{2,3,4,6,9\} \rightarrow (6)\end{aligned}$$

Consider the state C :

When input is a :

$$\begin{aligned}\delta(c,a) &= \text{ECLOSE}(\delta_E(c,a))\\ &= \text{ECLOSE}(\delta_E\{2,3,4,6,9\},a)\\ &= \text{ECLOSE}(5)\\ &= \{5,8,9,3,4,6\}\\ &= \{3,4,5,6,8,9\} \rightarrow (D)\end{aligned}$$

When input is b :

$$\begin{aligned}\delta(A,b) &= \text{ECLOSE}(\delta E(A,b))\\ &= \text{ECLOS}(\delta_E\{2,3,4,6,9\},b)\\ &= \text{ECLOSE}(\{7\})\\ &= \{7,8,9,3,4,6\}\\ &= \{3,4,6,7,8,9\} \rightarrow (E)\end{aligned}$$

Consider the state D :

When input is a:

$$\begin{aligned}\delta(D,a) &= \text{ECLOSE}(\delta_E(D,a))\\ &= \text{ECLOSE}(\delta_E\{3,4,5,6,8,9\},a)\\ &= \text{ECLOSE}(\{7\})\\ &= \{7,8,9,3,4,6\}\\ &= \{3,4,6,5,8,9\} \rightarrow (D)\end{aligned}$$

When input is b

$$\begin{aligned}\delta(B,b) &= \text{ECLOSE}(\delta_E(B,b))\\ &= \text{ECLOSE}(\delta E\{3,4,6,8,5,9\},b)\\ &= \text{ECLOSE}(7,8,9,3,4,6)\\ &= \{3,4,6,7,8,9\} \rightarrow (E)\end{aligned}$$

Consider the state E :

When input is a:

$$\begin{aligned}\delta(E,a) &= \text{ECLOSE}(\delta_E(E,a))\\ &= \text{ECLOSE}(\delta_E\{3,4,5,6,7,8,9\},a)\\ &= \text{ECLOSE}(\{S\})\\ &= \{5,8,9,3,4,6\}\\ &= \{3,4,5,6,8,9\} \rightarrow (D)\end{aligned}$$

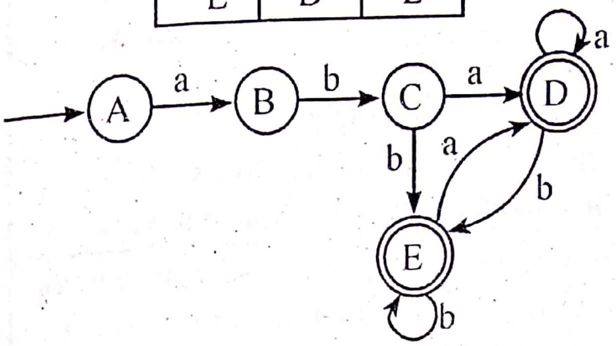
When input is b

$$\begin{aligned}\delta(E,b) &= \text{ECLOSE}(\delta_E(B,b))\\ &= \text{ECLOSE}(\delta_E\{3,4,6,7,8,5,9\},b)\\ &= \text{ECLOSE}(\{S\})\\ &= (7,8,9,3,4,6)\\ &= \{3,4,6,7,8,9\} \rightarrow (E)\end{aligned}$$

Since no new state, will stop

Since no new state, will stop

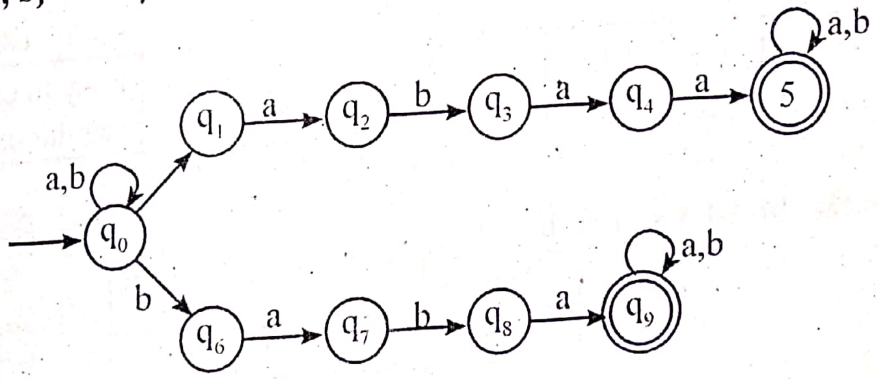
δ	a	b
A	B	ϕ
B	ϕ	C
*C	D	E
*D	D	E
*E	D	E



OR

2. a. Draw a DFSM to accept the language,
1 = {ω∈ {a, b}* : ∀x, y∈ {a, b}* ((ω = x abbaay) v (ω = x babay))} **(03 Marks)**

Ans.



b. Define distinguishable and indistinguishable states. Minimize the following DFSM,

S	0	1
A	B	A
B	A	C
C	D	B
*D	D	A
E	D	F
F	G	E
G	F	G
H	G	D

When k=2

$$R_{11}^{(2)} = R_{11}^{(1)} + R_{12}^{(1)}\left[R_{22}^{(1)}\right]^{*} R_{21}^{(1)}$$
$$= 1^{*} + 1^{*}0(\varepsilon + 11^{*}0)^{*}11^{*}$$
$$= 1^{*} + 1^{*}0(11^{*}0)^{*}11^{*}$$

$$R_{12}^{(2)} = R_{12}^{(1)} + R_{12}^{(1)}\left[R_{22}^{(1)}\right]*R_{22}^{(1)}$$
$$= 1*0 + 1*0(\varepsilon + 11*0)*(\varepsilon + 11*0)$$
$$= 1*0 + 1*0(11*0)*(\varepsilon + 11*0)$$

$$R_{13}^{(2)} = R_{13}^{(1)} + R_{12}^{(1)}\left[R_{22}^{(1)}\right]^{*} R_{23}^{(1)}$$
$$= \phi + 1*0(\varepsilon + 11*0)*0$$
$$= (0 + \varepsilon) + 1(11*0)*0$$

$$R_{21}^{(2)} = R_{21}^{(1)} + R_{22}^{(1)}\left[R_{22}^{(1)}\right]*R_{21}^{(1)}$$
$$= 11* + (\varepsilon + 11*0)(\varepsilon + 11*0)*11*$$
$$= 11* + (\varepsilon + 11*0)(11*0)11*$$

$$R_{22}^{(2)} = R_{22}^{(1)} + R_{22}^{(1)}\left[R_{22}^{(1)}\right]*R_{22}^{(1)}$$
$$= (\varepsilon + 11*0) + (\varepsilon + 11*0)(\varepsilon + 11*0)*(\varepsilon + 11*0)$$
$$= (\varepsilon + 11*0) + (\varepsilon + 11*0)(11*0)(\varepsilon + 11*0)$$

$$R_{31}^{(2)} = R_{31}^{(1)} + R_{32}^{(1)}\left[R_{22}^{(1)}\right]^{*} R_{21}^{(1)}$$
$$= \phi + 1(\varepsilon + 11*0)*11*$$
$$= 1(11*0)*11*$$

$$R_{32}^{(2)} = R_{32}^{(1)} + R_{32}^{(1)}\left[R_{22}^{(1)}\right]*R_{22}^{(1)}$$
$$= 1 + 1(\varepsilon + 11*0)*(\varepsilon + 11*0)$$
$$= 1 + 1(11*0)*(\varepsilon + 11*0)$$

$$R_{33}^{(2)} = R_{33}^{(1)} + R_{32}^{(1)}\left[R_{22}^{(1)}\right]^{*} R_{23}^{(1)}$$
$$= 1*0(11*0)*0$$

Final RE can be calculated as

$$R_{13}^{(3)} = R_{13}^{(2)} + R_{13}^{(2)}\left[R_{33}^{(2)}\right]*R_{33}^{(2)}$$
$$= 1*0(11*0)*0 + 1*0(11*0)*0\left[(0+\varepsilon) + 1(11*0)*0\right]*(0+\varepsilon) + 1(11*0)*0$$

b. Give Regular expressions for the following languages on /_= {a,b,c}
(i) all strings containing exactly one a
(ii) all strings containing no more than 3 a's.
(iii) all strings that contain at least one occurance of each symbol in V.

(03 Marks)

Ans.

(i) $R \in = (b+c)^{*}a(b+c)^{*}$

(ii) $R \in = (b+c)^{*}(\sigma + a)(b+c)^{*}(\varepsilon + a)(b+c)^{*}$

(iii) $(a+b+c)^{*}$

3. c. Let L be the language accepted by the following finite state machine. **(04 Marks)**

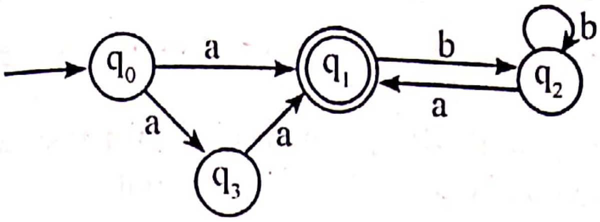


Fig. Q3 (c;

Indicate for each of the following regular expressions, whether it correctly describes L:

(i) (a U ba) bb* a
(ii) (ε U b) a (bb* a)*
(iii) ba U ab*a
(iv) (a U ba) (bb*a)*

Ans. i. NO
ii. YES
iii. NO
iv. YES

OR

4. a. Prove that the following language in not regular :
$L = \{0^n 1^n \mid \eta > 0\}$ **(05 Marks)**

Ans. Step 1 : Let L i.e, regular and η be the number of states
$x = 0^n b^n$

Step 2 : Since $|x| = 2\eta > \eta$ we can split x into uvw such that $|uv| \le \eta$ and $|V| \ge 1$ as

$$x = \overbrace{\underbrace{a\,a\,a\,a\,a\,a}_{u}\;\underbrace{a}_{V}}^{n}\;\underbrace{\overbrace{b\,b\,b\,b\,b\,b\,b\,b}^{n}}_{w}$$

Step 3 : According to pumping lemma $uv1\omega \in L$ for $i = 0,1,2.......$
When $i = 0$ 'V' doesn't exist so $L = \{0^n 1^n \mid n > 0\}$ is not regular

b. If L_1 and L_2 are regular languages then prove that $L_1 U L_2$, $L_1 . L_2$ and L_1* are regular languages. **(05 Marks)**

Ans. Refer Q.no.3(b) of MQP - 2.

c. Is the following grammar is ambiguous? **(06 Marks)**
$S \rightarrow i C t s | i c t \; s e s | a$
$C \rightarrow b$ **(06 Marks)**

Ans. Refer Q.no.5(b) of MQP - 2.

Module-3

5. a. Define Grammar, Derivation, Sentential forms and give one example for each. **(03 Marks)**

Ans. A grammer G is 4 tuple or quadruple G = (V,T,P,S) where 'V' is variable, T is terminals, P is production and 'S' is start symbol.

Ex: $S \rightarrow \varepsilon$, $S \rightarrow aS$

$A \Rightarrow \alpha \beta r$, the process of obtaining strings of terminal s and / or non - terminals from the start symbol by applying some or all productions is called derivation.

$$E \Rightarrow E + E, E \Rightarrow id + E, E \Rightarrow id + id$$

Let G = (V,T,P,S) be a grammar. The string ω obtained from the grammar G such that $S \Rightarrow w$ is called sentence of grammar G. Here, w is the string of terminals.

b. What is CNF? Obtain the following grammar in CNF **(09 Marks)**

$S \rightarrow ASB \mid \varepsilon$

$A \rightarrow aAS \mid a$

$B \rightarrow SbS \mid A \mid bb$

Ans. Let G = (V,T,P,S) be a CFG. The grammr G is said to be in CNF if all productions are of the form.

$A \rightarrow BC$ or $A \rightarrow a$

Eliminate ε - production

0v	nv	Production
ϕ	$S \rightarrow \varepsilon$	$S \rightarrow \varepsilon$
S	S	–

V = {A,B} are nullable variables

Production	Resulting production (p^1)
$S \rightarrow A S B$	$S \rightarrow AB$
$A \rightarrow aAS$	$A \rightarrow aA \mid a$
$B \rightarrow SbS$	$B \rightarrow SbS \mid bS \mid Sb \mid b \mid A \mid bb$

Given Production	Action	
$S \rightarrow AB$	Already in CNF	$S \rightarrow AB$
$A \rightarrow aA$	Replace a by A_0 $A_0 \rightarrow A$	$A \rightarrow A_0 A$ $A_0 \rightarrow a$ $A \rightarrow a$
$B \rightarrow SbS \mid bS \mid Sb \mid A$	Replace b by B_0 $B_0 \rightarrow b$	$B \rightarrow SB_0 S \mid B_0 S \mid SB_0 \mid a$

Replace B_0S with B_1

$B \rightarrow SB_1$

$B_1 \rightarrow B_0S$

$G' = (V', T, P', S)$ 'S' is start symbol

$V = (S, A, B, B_0, B_1, A_0)$

$P = \{$ $S \rightarrow AB$ $B \rightarrow SB_1 \mid B_0S \mid SB_0 \mid a \mid$

$A \rightarrow A_0A$ $B_0 \rightarrow b$

$A_0 \rightarrow a$ $B_1 \rightarrow B_0S$

$A \rightarrow a$

$\}$

c. Let G be the grammar,

$S \rightarrow aB \mid bA$

$A \rightarrow a \mid aS \mid bAA$

$B \rightarrow b \mid bS \mid aBB$

For the string qaabbabbba find a

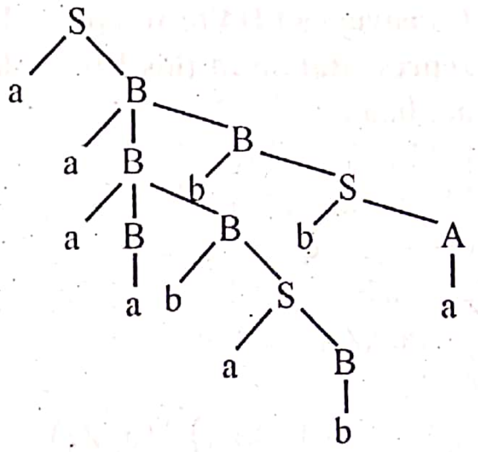
(i) Left most derivation.

(ii) Right most derivation.

(iii) Parse tree. **(04 Marks)**

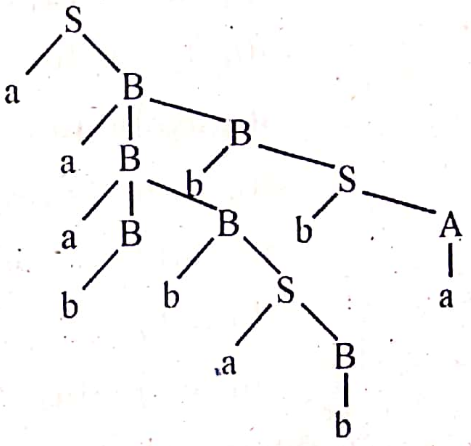
Ans. i. Let most derivation

$S \Rightarrow aB$
$\Rightarrow aaBB$
$\Rightarrow aaaBBB$
$\Rightarrow aaaabbSB$
$\Rightarrow aaaabbaBB$
$\Rightarrow aaaabbabB$
$\Rightarrow aaaabbabbS$
$\Rightarrow aaaabbabbbA$
$\Rightarrow aaaabbabbba$



ii. Right most derivation

$S \Rightarrow aB$
$\Rightarrow aaBB$
$\Rightarrow aaaBbS$
$\Rightarrow aaaBbbA$
$\Rightarrow aaaBbba$
$\Rightarrow aaaaBBbba$
$\Rightarrow aaaaBbSbba$
$\Rightarrow aaaBbaBbba$
$\Rightarrow aaaaBbabbba$
$\Rightarrow aaaabbabbba$



OR

6. a. Explain the following terms:
(i) Pushdown automata (PDA).
(ii) Languages of a PDA.
(iii) Instantaneous description of a PDA.

(03 Marks)

Ans. **i. Pushdown Automata (PDA) :** A PDA is a seven tuple

$M = (Q,\Sigma,|,\delta,q_0, Z_0, F)$

Q is set of finite states

Σ is set of input alphabets

| is set of stack alphabets

δ is transition Q x (Σ U ε) x | – Q x |*

q_0 ε Q is start state

Z_0 ε | is initial symbol on stack

F ≤ Q is set of final state

ii. Language of PDA : The language LCM accepted by a final state is defined as $L(M) = \{w \mid (q_0,w,Z_0) \mid {}^*(P,\varepsilon,\alpha)|$

iii. Instantaneous description : Let $M = (Q,\Sigma,|,\delta,q_0, Z_0, F)$ be a PDA. An ID (instantaneous description) is defined as 3-tuple or a triple (q,w,α)

b. Construct a PDA to accept the language $L=\{ww^R|w\varepsilon\{a,b\}^*\}$. Draw the graphical representation of this PDA. Show the moves made by this PDA for the string aabbaa.

(10 Marks)

Ans. $L(M) = \{ww^R|w\varepsilon\{a,b)^*\}$

$M = (Q,\Sigma,|,\delta,q_0, Z_0, F\}$

$Q = \{q_0,q_1,q_2\}$

$\Sigma = \{a,b\}$

$| = \{a,b,Z_0)$

δ : {

$$(q_0,\varepsilon,Z_0)=(q_{1,}Z_0)$$
$$\delta(q_0,a,Z_0)=(q_0,aZ_0)$$
$$\delta(q_0,b,Z_0)=(q_0,bZ_0)$$
$$\delta(q_0,b,a)=\{(q_0,aa),(q,\varepsilon)\}$$
$$\delta(q_0,a,b)=(q_0,ba)$$
$$\delta(q_0,a,b)=(q_0,ab)$$
$$\delta(q_0,a,b)=(q_0,ab)$$
$$\delta(q_0,b,b)=\{(q_0,bb),(q,\varepsilon)\}$$
$$\delta(q_1,a,a)=(q_1,\varepsilon)$$
$$\delta(q_1,b,b)=(q_1,\varepsilon)$$
$$\delta(q_1,\varepsilon,Z_0)=(q_0,Z_0)$$

}

$q_0 \in Q$ is start state

$Z_0 \in P$ is initial stack symbol

$F = \{q_2\}$ is final state

$$\begin{tikzcd}
(q_0, aabbaa, z_0) \arrow[d] & & & \\
(q_0, abbaa, az_0) \arrow[r] \arrow[d] & (q_0, aabbaa, z_0) \arrow[r] & (q_2, aabbaa, z_0) & \\
(q_0, bbaa, aaz_0) \arrow[r] \arrow[d] & (q_1, bbaa, z_0) \arrow[r] & (q_2, bbaa, z_0) & \\
(q_0, baa, baaz_0) \arrow[r] \arrow[d] & (q_0, aa, bbaaz_0) \arrow[r] & (q_0, a, abbaaz_0) \arrow[d] \arrow[ddr] & \\
(q_1, aa, aaz_0) \arrow[d] & & (q, \varepsilon, bbaaz_0) & \\
(q_1, a, az_0) \arrow[d] & & & (q_0, \varepsilon, abbaaz_0) \\
(q_1, \varepsilon, z_0) \arrow[r] & (q_2, \varepsilon, z_0) & &
\end{tikzcd}$$

c. **Convert the following CFG to PDA**

$S \to aABB|aAA$

$A \to aBB|a$

$B \to bBB\,|\,A$

$C \to a$

(03 Marks)

Ans. $Q = \{q_0, q_1, q_2\}$

$\Sigma = \{a, b\}$

$\Gamma = \{S, A, B, C_1, Z_0\}$

$\delta:\{$

$$\delta(q_0, \varepsilon, Z_0) = (q_1, SZ_0)$$
$$\delta(q_1, a, S) = (q_1, ABB)$$
$$\delta(q_1, a, S) = (q_1, AA)$$
$$\delta(q_1, a, A) = (q_1, BB)$$
$$\delta(q_1, a, A) = (q_1, \varepsilon)$$
$$\delta(q_1, b, B) = (q_1, BB)$$
$$\delta(q_1, a, B) = (q_1, BB)$$
$$\delta(q_1, a, B) = (q_1, \varepsilon)$$
$$\delta(q_1, a, c) = (q_1, \varepsilon)$$
$$\delta(q_1, \varepsilon, Z_0) = (q_f, Z_0)$$

$\}$

$q_0 \in Q$ is start state

$Z_0 \in T$ is stack symbol

$F = \{q_f\}$ is final state

Module-4

7. a If L_1 and L_2 are context free languages then prove that $L_1 U L_2$, $L_1 L_2$ and L_1^*, are context free languages. **(04 Marks)**

Ans.

$(i) G_1 = (V_1, T_1, P_1, S_1)$

$G_2 = (V_2, T_2, P_2, S_2)$

$G_3 = (V_1 U V_2 U S_3, T_1 U T_2, P_3, S_3)$

S_3 is a start state G_3 and $S_3 \in (V_1 U V_2)$

$P_3 = P_1 U P_2 U \{S_3 \rightarrow S_1 / S_2\}$

$L_3 = L_1 U L_2$

$(ii) G_4 = (V_1 U V_2 U S_4, T_1 U T_2, P_4, S_4)$

S_4 is a start symbol for the grammar G_4 and $S_4 \in (V_1 U V_2)$

$P_4 = P_1 U P_2 U \{S_4 \rightarrow S_1 S_2\}$

$\therefore L_3 = L_1 . L_2$

$(iii) G_5 = (V, U S_5, T_1, P_5, S_5)$

S_5 is a the start symbol of Grammar G_5

$P_5 = P_1 U \{S_S \rightarrow S_1 S_5 | \varepsilon\}$

$T L_5 = L_S^*$

b. Give a decision procedure to answer each of the following questions:

(i) Given a regular expression a and a PDA M, the language accepted by M a subset of the language generated by a?

(ii) Given a context-free Grammar G and two strings Si and S2, does G generate $S_1 S_2$?

(iii) Given a context free Grammar G, does G generate any even length strings.

(iv) Given a Regular Grammar G, is L(G) context-free? **(12 Marks)**

Ans. i. Observe that this is true if $\perp(M) \cap L(\alpha) = \phi$. So the following procedure answers the question :

1. From α, build a PDA M^* so that $L(M^*) = L(\alpha)$
2. From M and M^*, build a PDA M^{**} that accepts $L(M) \cap L(M^*)$
3. If $L(M^{**})$ is empty , return true else return false.

ii. 1. Convert G to chomsky normal forms2. Try all derivations in G of length up to $2|S_1 S_2|$. If any of them generates $S_1 S_2$ return True, else return false

iii. 1. Use CFG to PDA topolown (G) to build a PDA P hta taccepts L(G).
2. Build an FSM E that accepts all even length strings over the alphabet Σ_G.
3. Use insert PDA and FSM(P,E) to build a PDA P* that accepts $L(G) \cap L(E)$.
4. Return decioleCFLempty(P*)

iv. i. Return True (Since every regular language is context free)

OR

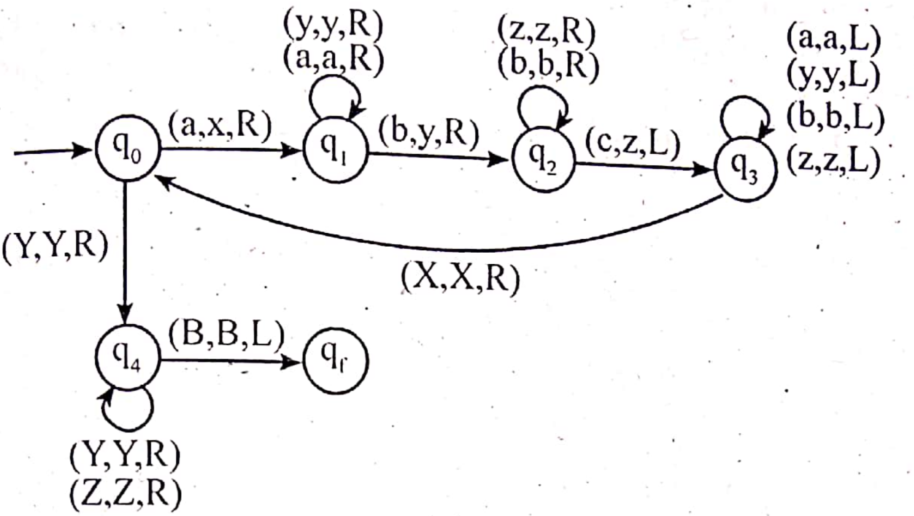
8. a. **Explain with neat diagram, the working of a Turing Machine model. (05 Marks)**

Ans. Refer Q.no.9(a) of MQP - 2.

b. **Design a Turing machine to accept the language L= {a^n b^n c^n | η >= 1}. Draw the transition diagram. Show the moves made by this turing machine for the string aabbcc.**

(11 Marks)

Ans.



a a b b c c

x a b b c c

x a b b c c

x a y b c c

x a y b c c

x a y b z c

x x y b z c

x x y y z c

x x y y z z

Module-5

9 **Write short notes on:**

a. Multi-tape turning machine.

b. Non-deterministic turning machine.

c. Linear Bounded automata.

(16 Marks)

Ans. a. Refer Q.no. 9(b) of MQP - 1.

b. **Non - deterministic turning machine :** In a non - deterministic turning machine, for every state and symbol, there are a group of actions the TM can have. So here the transitions are not deterministic. The computation of a non - deterministic turning machine is a tree of configurations that can be reached from the start configuration. An input is accepted if there is at least one node of the tree which is an accept configuration, otherwise it is not accepted. If all branches of the computational tree hact on all inputs, the non - deterministic turning machine is called a decideg and if for some input, all branches are rejected, the input is also rejected.

c. Linear bounded automata : Refer Q.no. 10(b) of MQP - 1.

OR

10. Write short notes on:

a. Undecidable languages.

b. Halting problem of turning machine.

c. The post correspondence problem.

(16 Marks)

Ans. a. Refer Q.no.9(b) of MQP - 2

b. Refer Q.no.10(a) of MQP - 1

c. Refer Q.no.10b(i) of MQP - 2.

Fifth Semester B.E. Degree Examination, CBCS - June / July 2018

Automata Theory & Compatibility

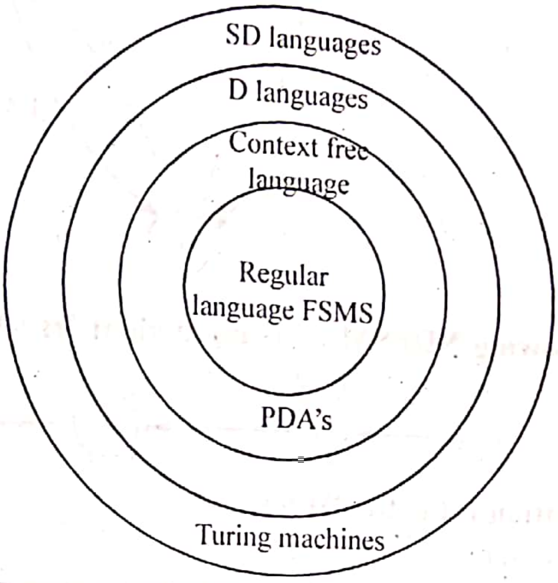
Time: 3 hrs. Max. Marks: 80

Note : *Answer any FIVE full questions, selecting ONE full question from each module.*

Module - 1

1. a. **With a neat diagram, explain a hierarchy of language classes in automata theory.** **(04 Marks)**

Ans.



Grammar	Language	Automaton
Type - 0	Recursively enumerable	Turing machine
Type - 7	Context sensitive	Linear bounded Non - deterministic Turing machine
Type - 2	Context free	Non determinstic pushdown automata
Type - 3	Regular	finite state automaton

b. **Define deterministic FSM. Draw a DFSM to accept decimal strings which are divisible by 3.** **(06 Marks)**

Ans. Step 1 :- d = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9} K = 3

Step 2 :- After dividing by 3, possible reminder are 0, 1, 2

Step 3 :- Compute transition

$$\delta(q_i, a) = q_j \quad \text{where } j = (r * i + d) \bmod K$$

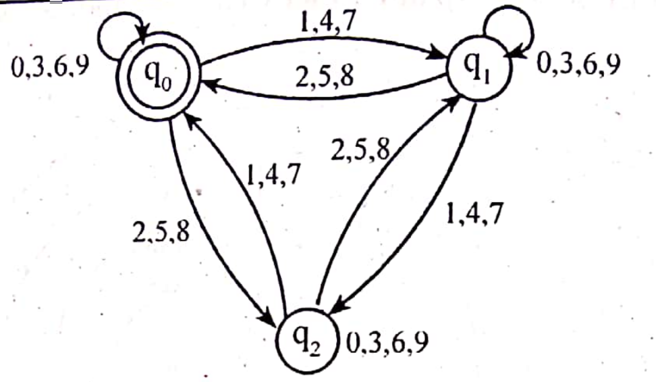
wih r = 10 and K = 3

{0, 3, 6, 9} leaves 0 as reminder)

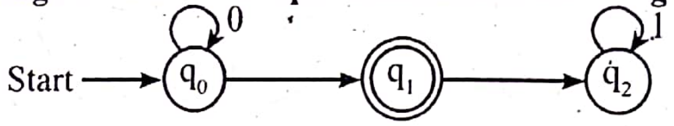
{1, 4, 7} leaves 1 as reminder)

{2, 5, 8} leaves 2 as reminder)

i = 0	0 1 2	(10 * 0 + 0) Mod 3 = 0 (10 * 0 + 1) Mod 3 = 1 (10 * 0 + 2) Mod 3 = 2	$\delta(q_0, 0) = q_0$ $\delta(q_0, 1) = q_1$ $\delta(q_0, 2) = q_2$	$\delta(q_0, \{0, 3, 6, 9\}) = q_0$ $\delta(q_0, \{1, 4, 7\}) = q_1$ $\delta(q_0, \{2, 5, 8\}) = q_2$
i = 1	0 1 2	(10 * 1 + 0) Mod 3 = 1 (10 * 1 + 1) Mod 3 = 2 (10 * 1 + 2) Mod 3 = 0	$\delta(q_1, 0) = q_1$ $\delta(q_1, 1) = q_2$ $\delta(q_1, 2) = q_0$	$\delta(q_1, \{0, 3, 6, 9\}) = q_1$ $\delta(q_1, \{1, 4, 7\}) = q_2$ $\delta(q_1, \{2, 5, 8\}) = q_0$
i = 2	0 1 2	(10 * 2 + 0) Mod 3 = 2 (10 * 2 + 1) Mod 3 = 1 (10 * 2 + 2) Mod 3 = 0	$\delta(q_2, 0) = q_2$ $\delta(q_2, 1) = q_0$ $\delta(q_2, 2) = q_1$	$\delta(q_2, \{0, 3, 6, 9\}) = q_2$ $\delta(q_2, \{1, 4, 7\}) = q_0$ $\delta(q_2, \{2, 5, 8\}) = q_1$



c. Convert the following NDFSM to its equivalent DFSM Rfer Fig 1.c.



Also write transition table for DFSM (06 Marks)

Ans. **Step 1 :** - Identify start state $Q_D = \{q_0\}$

Step 2 :- Identify alphabet $\Sigma = \{0, 1\}$

Step 3 :- Transitions

Input symbol = 0

$\delta_D = (\{q_0\}, 0) = \delta_N (\{q_0\}, 0) = \{q_0, q_1\}$

For state $\{q_0, q_1\}$

$\delta_D = (\{q_0, q_1\}, 0) = \delta_N (\{q_0, q_1\}, 0)$

$= \delta_N \{q_0, 0\} \cup \delta_N(q_1, 0)$

$= \{q_0, q_1\} \cup \{q_2\} = \{q_0, q_1, q_2\}$

For state $\{q_1\}$

Input Symbol = 0

$\delta_D = (\{q_1\}, 0) = \delta_N (\{q_1\}, 0)$

For state $\{q_0, q_1, q_2\}$

Input Symbol = 0

$\delta_D = (\{q_0, q_1, q_2\}, 0) = \delta_N (\{q_0, q_{1,} q_2\}, 0)$

$= \delta_N \{q_0, 0\} \cup \delta_N(q_1, 0) \cup \delta_N(q_2, 0)$

$= \{q_0, q_1\} \cup \{q_2\} \{\phi\} = \{q_0, q_1, q_2\}$

For state $\{q_1, q_2\}$

$\delta_D = (\{q_1, q_2\}; 0) = \delta_N (\{q_1, q_2\}, 0)$

$= \delta_N \{q_1, 0\} \cup \delta_N(q_2; 0)$

$= \{q_2\} \cup \phi = \{q_2\}$

For state $\{q_2\}$

Input Symbol = 0

$\delta_D = (\{q_2\}, 0) = \delta_N (\{q_2\}, 0) = \{\phi\}$

Input symbol = 1

$\delta_D = (\{q_0\}, 1) = \delta_N (\{q_0\}, 1) = \{q_1\}$

$\delta_D = (\{q_0, q_1\}, 1) = \delta_N (\{q_0, q_1\}, 1)$

$= \delta_N \{q_0, 1) \cup \delta_N(q_1, 1\}$

$= \{q_1\} \cup \{q_2\} = \{q_1, q_2\}$

Input Symbol = 1

$\delta_D = (\{q_1\}, 1) = \delta_N (\{q_1\}, 1)$

Input Symbol = 1

$\delta_D = (\{q_0, q_1, q_2\}, 1) = \delta_N (\{q_0, q_1, q_2\}, 1)$

$= \delta_N \{q_0, 1) \cup \delta_N(q_1, 1\} \cup \delta_N(q_2, 1\}$

$= \{q_1\} \cup \{q_2\} \cup \{q_2\} = \{q_1, q_2\}$

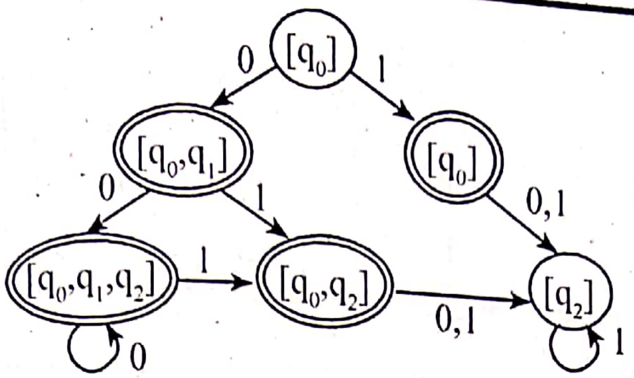
$\delta_D = (\{q_0, q_1\}, 1) = \delta_N (\{q_0, q_1\}; 1)$

$= \delta_N \{q_1, 1) \cup \delta_N(q_2, 1\}$

$= \{q_2\} \cup \{q_2\} = \{q_2\}$

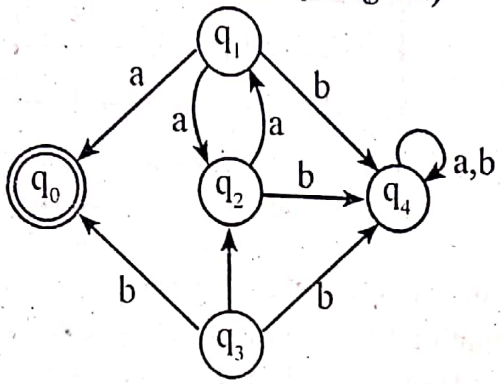
Input Symbol = 1

$\delta_D = (\{q_2\}, 1) = \delta_N (\{q_2\}, 1) = \{q_2\}$



OR

Minimize the following finite automata (Refer Fig 2a.) (06 Marks)



Step 1 :-

q_1				
q_2				
q_3				
q_4	X	X	X	X
	q_0	q_1	q_2	q_3

Step 2 :-

δ	a	b
(p, q)	(r, s)	(r, s)
(q_0, q_1)	(q_1, q_2)	(q_3, q_4)
(q_0, q_2)	(q_1, q_1)	(q_3, q_4)
(q_0, q_3)	(q_1, q_2)	(q_3, q_4)
(q_1, q_2)	(q_2, q_1)	(q_4, q_4)
(q_1, q_3)	(q_2, q_2)	(q_4, q_4)
(q_2, q_3)	(q_1, q_2)	(q_4, q_4)

q_1	X			
q_2	X			
q_3	X			
q_4	X	X	X	X
	q_0	q_1	q_2	q_3

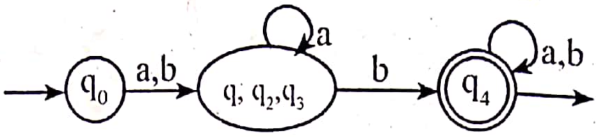
Step 3 :-

δ	a	b
(p, q)	(r, s)	(r, s)
(q_1, q_2)	(q_2, q_1)	(q_4, q_4)
(q_1, q_3)	(q_2, q_2)	(q_4, q_4)
(q_2, q_3)	(q_1, q_2)	(q_4, q_4)

None of the unmarked pairs (r, s) are marked in table.

Step 4 :- (q_0) $(q_1\ q_2\ q_3)$ (q_4)

State	a	b
q_0	$(q_1\ q_2\ q_3)$	$(q_1\ q_2\ q_3)$
$(q_1\ q_2\ q_3)$	$(q_1\ q_2\ q_3)$	(q_4)
(q_4)	(q_4)	(q_4)



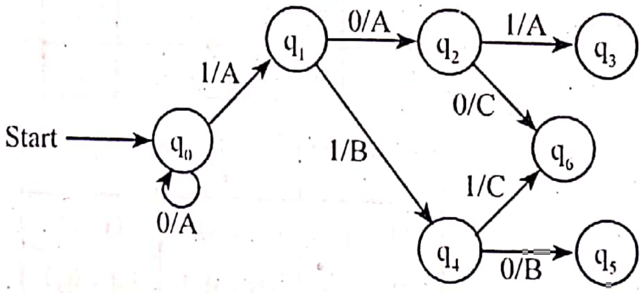
b. Construct a mealy machine for the following

i) Design a mealy machine for a binary input sequence. Such that if it has substring 101, the machine outputs A. if input has substring 110, the machine outputs. B otherwise it outputs C.

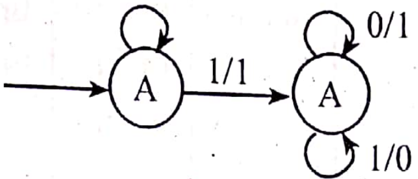
ii) Design a mealy machine that takes binary number as input and produces 2's complement of that number as input.

Assume the string is read from LSB to MSB and end carry is discarded. (06 Marks)

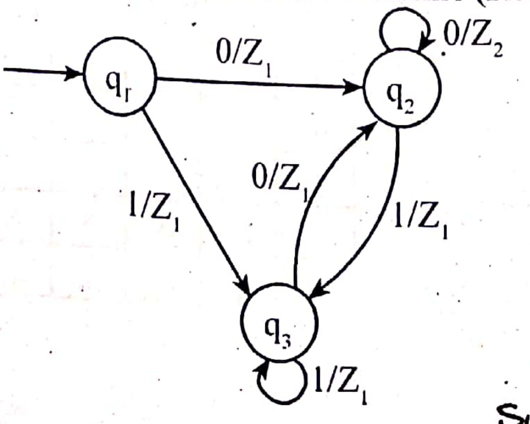
Ans. i)



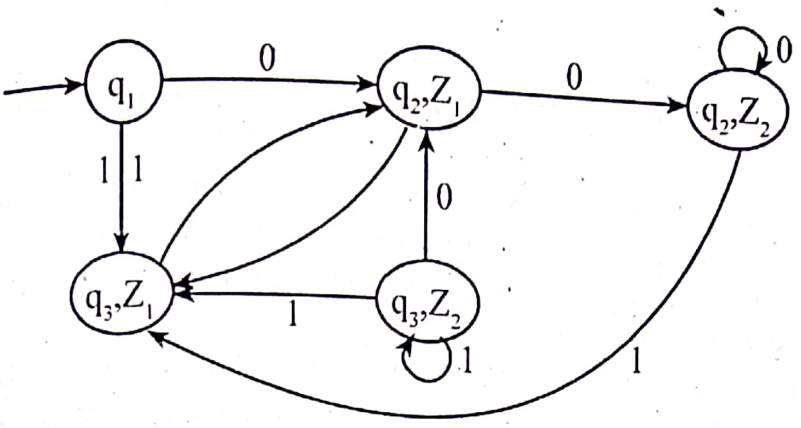
ii)



c. Convert the following mealy machine to moore machine (Refer fig 2.c.) (04 Marks)



Ans.



Module - 2

3. a. **Define regular expression. Obtain a regular expression for the following language:**

i) $L = \{a^n b^m \mid m + n \text{ is even}\}$

ii) $L = \{a^n b^m \mid m \geq 1\ n \geq 1, nm \geq 3\}$

iii) $L = \{ W : |W| \bmod 3 = 0 \text{ where } W \in \{a, b\}^*\}$ **(08 Marks)**

Ans. Definition :- Refer Q.No. 3.a. of MQP - 1

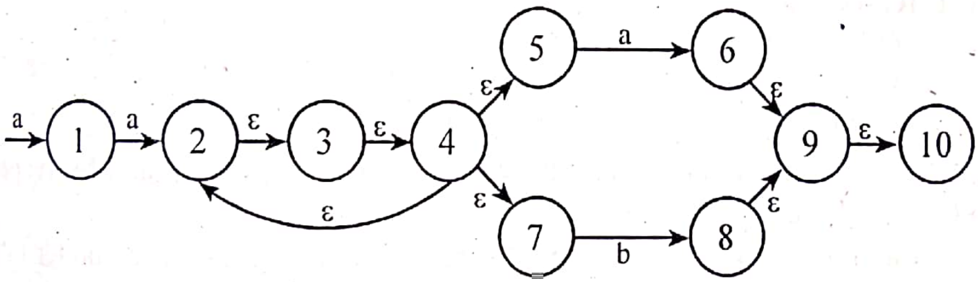
i) $R\epsilon = ((a+b)\ (a + b)$ * or $R\epsilon = (a\ a)^*\ (b\ b)^* + a(a\ a)^*\ b(b\ b)^*$

ii) $R\epsilon = aaaa^*b + abbbb^* + aaa^*\ bbb^*$

iii) $R\epsilon = ((a + b)\ (a + b)\ (a + b))^*$

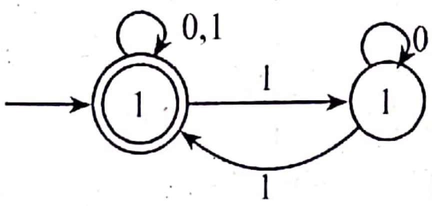
b. **Design an NDFSM that accept the language L(aa*(a + b))** **(04 Marks)**

Ans.



c. **Convert the regular expression (0 + 1) *'1(0 + 1) to NDFSM** **(04 Marks)**

Ans.



OR

4. a. **If the regular grammars define exactly the regular language, then prove that the class of languages that can be defined with regular grammars is exactly the regular languages.** **(04 Marks)**

Ans. We first show that any languages that can be defined with a regular grammar can be accepted by some FSM and so is regular. Then we must show that every regular language can be defined with a regular grammar. Both proofs are by construction.

Regular grammar → FSM : The following algorithm construct an FSM M from a regular grammar G = (V, Σ, R,, S) and assures that LCM) = LCG) :

grammar to FSM (G : regular grammar) =

1. Create in M a separate state for each non terminal in V.
2. Make the state corresponding to S the start state.
3. If there are any rules in R of the form x → w, for some w ∈ Σ, then create an additional state labeled #.
4. For each rule of the form x → wy, add a transition from z to y labeled w.
5. For each rule of the form x → w, add a transition from x to # labeled w.
6. For each rule of the form x → ε, add a transition from x as accepting.
7. Mark state # as accepting.
8. If M is in complete, M requires a dead state. Add a new state D. For every (q, i) pair for which no transition has already been defined, create a transition from q to D labeled i. for every i in Σ, create a transition from D to D labeled i.

b. Prove that the regular language are closed under complement, intersection, difference, reverse and letter substitution. **(08 Marks)**

Ans. **i) Under complement :-** Let M1 = $(Q, \Sigma, \delta, q_0, F)$ be a DFA which accepts the language 1. Since the language is accepted by a DFA, the language is regular. Noe let us define the machine $M_2 = (Q, \Sigma, \delta, q_0, Q - F)$ which accepts I. Note that there in no difference between M_1 and M_2 except the final states.

The non - final states of M_1 = are th final state of M_2 and final stat of M_1 are the non final states of M_2 so the language which is rejected by M_1 is accepted by M_2 and vice versa. Thus we have a machine M_2 which accepts all those strings denoted by I that are rejected by machine M_1. So regular language is closed under complement.

ii) Intersection :-

$M_1 = (Q, \Sigma, \delta_1, q_1, F_1)$ which accepts L_1

$M_2 = (Q, \Sigma, \delta_2, q_2, F_2)$ which accepts L_2

$Q = Q_1 \times Q_2$

$q = (q_1, q_2)$ where q_1 and q_2 are the start states of machine M_1 and M_2 respectively.

$\hat{\delta}_1(q_1,w) \in F_1$ and $\hat{\delta}_2(q_2,w) \in F_2$.

i.e..,if and only if $w \in L_1 \cap L_2$. So the regular language is closed under intersection.

iii) Difference

$M_1 = (Q, \Sigma, \delta_1, q_1, F_1) \rightarrow L_1$

$M_2 = (Q, \Sigma, \delta_2, q_2, F_2) \rightarrow L_2$

$\hat{\delta}_1(q_1,q_2),w)$ is in F

$(\hat{\delta}_1(q_1,w) \in F_1$ and $(\hat{\delta}_2(q_2,w) \in F_2$

i.e., regular language is close under difference.

iv) Reversal and letter substitution

$L(E^R) = (L(E))^R$

Refer Q.No. 3b. of MQP - 2.

c. State and prove pumping lemma for regular language. **(04 Marks)**

Ans. Refer Q.No. 3b. of MQP - 1.

Module - 3

5. a. Define a context free grammar. Obtain the grammar to generate the language L= {W} $n_a(w) = n_b(w)$} **(04 Marks)**

Ans. S → ε

S → a s b

$S \to b\,s\,a$

$G = (V, J, P, S)$

$V = \{S\}$

$T = \{a, b\}$

$P = \{\delta \to \varepsilon$

$\delta \to a\,s\,b$

$\delta \to b\,s\,a$

$\}$

S is start symbol

b. **For the regular expression (011 + 1)* (01)* obtain the context free grammar.**

(04 Marks)

Ans. $G = (V, T, P, S)$

$V = \{S, A\}$

$T = \{0, 1\}$

$P = \{$

$S \to \varepsilon \mid 0\,1\,1\,A \mid 0\,1\,S$

$A \to 1\,S \mid \varepsilon$

$\}$

's' is start symbol.

c. **What is ambiguity? Show that the following grammar is ambigugous.**

$S \to aB \mid bA$

$A \to as \mid bAA \mid a$

$B \to bs \mid aBB \mid b.$

(08 Marks)

Ans. A grammar G is ambiguous if and only if there exists atleast one string $w \in T *$ ϕop which two or more different parse trees exist by applying either LMD or RMD.

$S \Rightarrow aB$

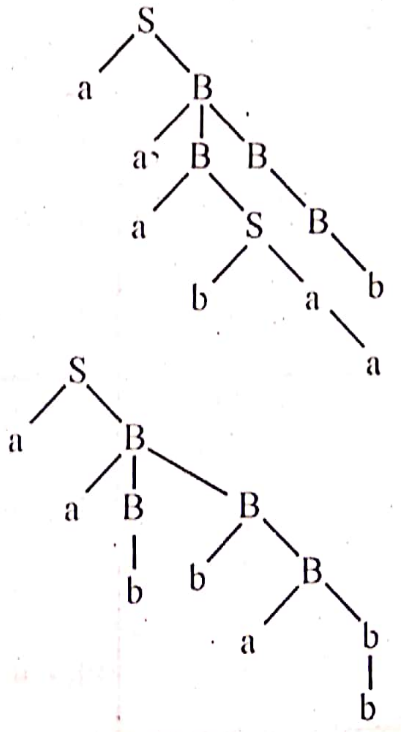
$S \Rightarrow aaBB$

$S \Rightarrow aabSB$

$S \Rightarrow aabbAB$

$S \Rightarrow aabbaB$

$S \Rightarrow aabbab$



$S \Rightarrow aB$

$S \Rightarrow aaBB$

$S \Rightarrow aabB$

$S \Rightarrow aabbS$

$S \Rightarrow aabbaB$

$S \Rightarrow aabbab$

Two parse tree for same string

So grammar is ambiguous.

OR

6. a. Define PDA. Obtain to accept the language 1(M) = {w C w^R) | w ∈ (a, b)8}, wher w^R is reverse of w by a final state. (08 Mark

Ans. Refer Q.No. 6.. of MQP - 1

b. For the grammar

S → aABB | aAA
A → aBB | a
B → bBB | A
C → a

Obtain the corresponding PDA. (04 Mark

Ans. Refer Q.No. 6.c. of Dec 2017 / Jan 2018

c. Obtain a CFG for the FDA shown below:

$f(q_0, a, Z) = (q_0, AZ)$
$f(q_0, a, A) = (q_0, A)$
$f(q_0, b, A) =(q, \varepsilon)$
$f(q_1, \varepsilon, Z) = (q_2, \varepsilon)$. (04 Mark

Ans.

For δ of the form $\delta(q_i, a, z) = (q_j, \varepsilon)$	Resulting production $(q_i\, Z\, q_j) \to a$
$\delta(q_0, a, A) = (q_3, \varepsilon)$ $\delta(q_0, b, A) = (q_1, \varepsilon)$ $\delta(q_1, \varepsilon, z) = (q_2, \varepsilon)$	$(q_0\, A\, q_3) \to a$ $(q_0\, A\, q_1) \to b$ $(q_1\, Z\, q_2) \to \varepsilon$

For δ of the form $\delta(q_i, a, z) = (q_j, \varepsilon)$	Resulting production $(q_i\, Z\, q_j) \to a$
$\delta(q_0, a, A) = (q_3, \varepsilon)$	$(q_0\, Z\, q_0) \to a(q_0\, A\, q_0)\, (q_0\, Z\, q_0) \mid a(q_0\, A\, q_1)\, (q_1\, Z\, q_0) \mid$ $a(q_0\, A\, q_2)\, (q_2\, Z\, q_0) \mid a(q_0\, A\, q_3)\, (q_3\, Z\, q_0)$ $(q_0\, Z\, q_1) \to a(q_0\, A\, q_0)\, (q_0\, Z\, q_1) \mid a(q_0\, A\, q_1)\, (q_1\, Z\, q_1) \mid$ $a(q_0\, A\, q_2)\, (q_2\, Z\, q_1) \mid a(q_0\, A\, q_3)\, (q_3\, Z\, q_1)$ $(q_0\, Z\, q_2) \to a(q_0\, A\, q_0)\, (q_0\, Z\, q_2) \mid a(q_0\, A\, q_1)\, (q_1\, Z\, q_2) \mid$ $a(q_0\, A\, q_2)\, (q_2\, Z\, q_2) \mid a(q_0\, A\, q_3)\, (q_3\, Z\, q_2)$ $(q_0\, Z\, q_3) \to a(q_0\, A\, q_0)\, (q_0\, Z\, q_3) \mid a(q_0\, A\, q_1)\, (q_1\, Z\, q_3) \mid$ $a(q_0\, A\, q_2)\, (q_2\, Z\, q_3) \mid a(q_0\, A\, q_3)\, (q_3\, Z\, q_3)$
$\delta(q_3, \varepsilon, z) = (q_0, Az)$	$(q_3\, Z\, q_0) \to (q_0\, A\, q_0)\, (q_0\, Z\, q_0) \mid (q_0\, A\, q_1)\, (q_1\, Z\, q_0) \mid$ $(q_0\, A\, q_2)\, (q_2\, Z\, q_0) \mid (q_0\, A\, q_3)\, (q_3\, Z\, q_0)$ $(q_3\, Z\, q_1) \to (q_0\, A\, q_0)\, (q_0\, Z\, q_1) \mid (q_0\, A\, q_1)\, (q_1\, Z\, q_1) \mid$ $(q_0\, A\, q_2)\, (q_2\, Z\, q_1) \mid (q_0\, A\, q_3)\, (q_3\, Z\, q_1)$ $(q_3\, Z\, q_2) \to (q_0\, A\, q_0)\, (q_0\, Z\, q_2) \mid (q_0\, A\, q_1)\, (q_1\, Z\, q_2) \mid$ $(q_0\, A\, q_2)\, (q_2\, Z\, q_2) \mid (q_0\, A\, q_3)\, (q_3\, Z\, q_2)$ $(q_3\, Z\, q_3) \to (q_0\, A\, q_0)\, (q_0\, Z\, q_3) \mid (q_0\, A\, q_1)\, (q_1\, Z\, q_3) \mid$ $(q_0\, A\, q_2)\, (q_2\, Z\, q_3) \mid (q_0\, A\, q_3)\, (q_3\, Z\, q_3)$

Module-4

7. a. Consider the grammar

$S \rightarrow 0A|1B$

$A \rightarrow 0AA|1S|1$

$B \rightarrow 1BB|0S|0$

Obtain the grammar in CNF. **(08 Marks)**

Ans. $A \rightarrow 1$

$B \rightarrow 0$

Given production	Action	Resulting production
$S \rightarrow 0A \mid 1B$	Replace 0 by B_0 $B_0 \rightarrow 0$ Replace 1 by B_1 $B_1 \rightarrow 1$	$S \rightarrow B_0A \mid B_1B$ $B_0 \rightarrow 0$ $B_1 \rightarrow 1$
$A \rightarrow 0AA \mid 1S$	Replace 0 by B_0 $B_0 \rightarrow 0$ Replace 1 by B_1 $B_1 \rightarrow 1$	$A \rightarrow B_0AA \mid B_1S$ $B_0 \rightarrow 0$ $B_1 \rightarrow 1$
$B \rightarrow 1BB \mid 0S$	Replace 0 by B_0 $B_0 \rightarrow 0$ Replace 1 by B_1 $B_1 \rightarrow 1$	$B \rightarrow B_1BB \mid B_0S$ $B_1 \rightarrow 1$ $B_0 \rightarrow 0$

Consider $A \rightarrow B_0AA$ and $B \rightarrow B_1BB$

$A \rightarrow B_0AA \Rightarrow A \rightarrow B_0D_1$

$D_1 \rightarrow AA$

$B \rightarrow B_1BB \Rightarrow B \rightarrow B_1D_2$

$D_2 \rightarrow BB$

$G^1 = (V^1, T, P^1, S)$ is in CNF where

$V_1 = \{S, A, B, B_0, B_1, D_1, D_2\}$

$T_1 = \{0, 1\}$

$P_1 = \{$

$S \rightarrow B_0 A | B_1B$

$A \rightarrow B_1S | 1| B_0D_1$

$B \rightarrow B_0S | 1| B_1D_2$

$B_0 \rightarrow 0$

$B_1 \rightarrow 1$

$D_1 \rightarrow AA$

$D_2 \rightarrow BB$

$\}$

'S' is the start symbol

b. **Show that $L = \{a^n b^n c^n \mid \eta \geq 0\}$ is not context free.** **(08 Marks)**

Ans. Refer Q.No. 8.b. of MQP - 2

OR

8. a. With a neat diagram, explain the working of a basic Turing machine. (04 Marks)

Ans. Refer Q.No. 9.a. of MQP - 2

b. Obtain a Turing machine to accept the language L = {0ⁿ1ⁿ | η ≥ 1}. (08 Marks)

Ans. Refer Q.No. 9.a. of MQP - 1

c. Briefly explain the techniques for TM construction. (04 Marks)

Ans. Refer Q.No. 9.b.(i) of MQP - 1

Module-5

9. a. Obtain a Turing machine to recognize the language $L = \{0^n 1^n 2^n \mid \eta \geq 1\}$. (08 Marks)

Ans.

States	0	1	2	Z	Y	X	B
q_0	q_1, x, R				q_4, Y, R		
q_1	q_1, 0, R	q_2, Y, R			q_1, Y, R		
q_2		q_2, 1, R	q_3, Z, L	q_3, Z, R			
q_3	q_3, 0, L	q_3, 1, L		q_3, Z, L	q_3, Y, L	q_0, X, R	
q_4				q_5, Z, R	q_4, Y, R		
q_5				q_5, Z, R			q_6, B, R
q_6							

b. Prove that $HALT_{TM}$ ={(M,W)| the Turing machine M halts on input W} is undecidable. (04 Marks)

Ans. Refer Q.No. 10.a. of MQP - 1

c. With example, explain the quantum computation. (04 Marks)

Ans. Refer Q.No. 10.b.(ii) of MQP - 2

OR

10 Write a short note on:

a. Multiple Turing machine

b. Non deterministic Turing machine

c. The model of linear bounded automaton

d. The post correspondence problem. (16 Marks)

Ans. a) Refer Q.No. 9.b.(i) of MQP - 1

b) Refer Q.No. 9.b. of Dec 2017 / Jan 2018

c) Refer Q.No. 10.b. of MQP - 1

d) Refer Q.No. 10.b. of MQP - 2

Fifth Semester B.E. Degree Examination, CBCS - Dec 2018 / Jan 2019

Automata Theory and Compatibility

Time: 3 hrs. Max. Marks: 80

Note : *Answer any FIVE full questions, selecting ONE full question from each module.*

Module - 1

1\. a. **Define the following with example :**
i) String ii) Language iii) Alphabet iv) DFSM. **(08 Marks)**

Ans. i) The sequence of symbols obtained from the alphabets of a language is calle a string. Formally, a string is defined as a finite sequence of symbols from the alphabet Σ.
Sequence of symbols from the alphabet Σ.
Ex : $\Sigma = \{0,1\}$

ii) A language can be defined as a set of strings obtained from Σ* where Σ is set of alphabet of a particular language. In others words, a language is subset of Σ* which is denoted by $L \subseteq \Sigma^*$.
Ex : $\{\varepsilon, 0, 1, 01, 10, 1100, 0011 \ldots\}$

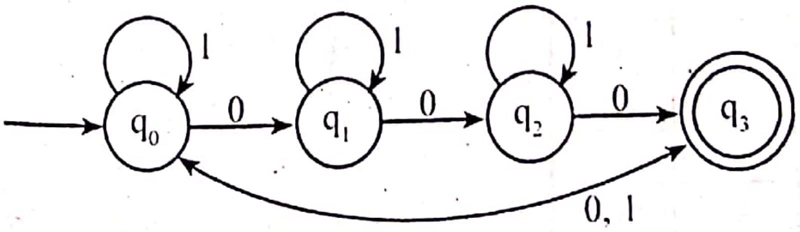
iii) A language consist of various symbols from which the words, statements etc., can be obtained. These symbols are called alphabets.
Ex : $\Sigma = \{a, b, \ldots, z, A, B, C, \ldots, Z, \#, \{, \}, (,) \ldots 0, \ldots 1\}$

iv) Deterministic Finite Automata (DFSM) is 5 - tuple or quintuple indicating five components $M = (Q, \Sigma, \delta, q_0, F)$

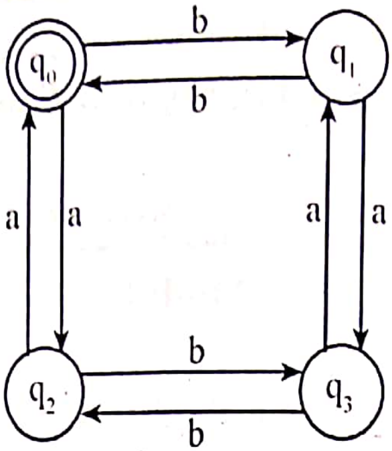
Where M is the name of machine
Q is non - empty finite set of states
Σ is non - empty finite set of input alphabets
δ is transition function $Q \times \Sigma \rightarrow Q$
$q_0 \in Q$ is start state
$F \subseteq Q$ is accepting or final states

b. **Design a DFSM to accept each of the following languages :**
i) $L = \{W \in \{0, 1\}^* : W$ has 001 as a substring$\}$
ii) $L = \{W \in \{a, b\}^* : W$ has even number of a's and even number of b's$\}$.
(08 Marks)

Ans. i)

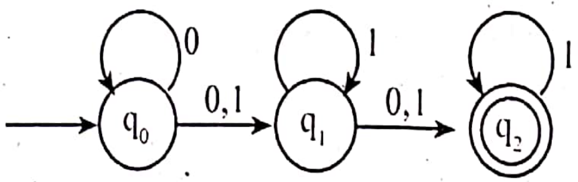


ii)



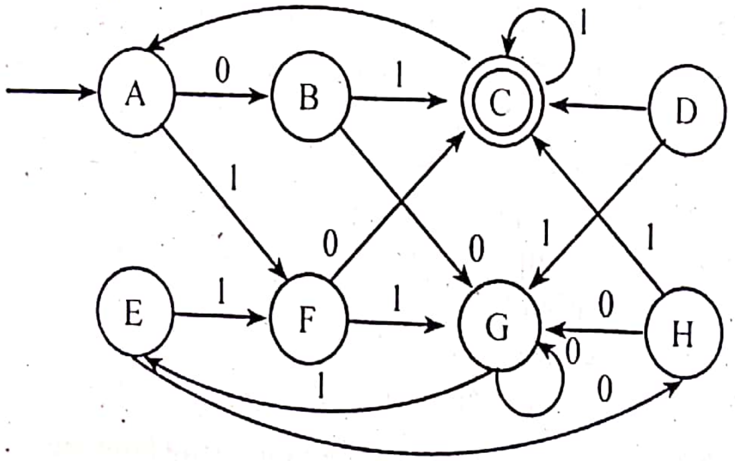
OR

2\. a. **Define NDFSM. Convert the following NDFSM to its equivalent DFSM.** (08 Marks)



Ans. Refer Q.no.1(c) of June/July 2018.

b. **Minimize the following DFSM.** (08 Marks)



Ans.

	0	1
→A	B	F
B	G	C
*C	A	C
D	C	G
E	H	F
F	C	G
G	G	E
H	G	C

Step 1 :

B							
C	X	X					
D		X					
E		X					
F		X					
G		X					
H		X					
	A	B	C	D	E	F	G

δ	a	b
(A,B)	(B,G)	(F,C)
(A,D)	(B,C)	(F,G)
(A,E)	(B,H)	(F,F)
A,F)	(B,C)	(F,G)
(A,G)	(B,G)	(F,E)
(A,H)	(B,G)	(F,C)
(B,D)	(G,C)	(C,G)
(B,E)	(G,H)	(C,F)
(B,F)	(G,C)	(C,G)
(B,G)	(G,C)	(C,E)
(B,H)	(G,G)	(C,C)
(D,E)	(C,H)	(G,F)
(D,F)	(C,C)	(G,G)
(D,G)	(C,G)	(G,E)
(D,H)	(C,G)	(G,C)
(E,F)	(H,C)	(F,G)
(E,G)	(H,G)	(F,E)
(E,H)	(H,G)	(F,C)
(F,G)	(C,G)	(G,E)
(F,H)	(G,G)	(E,C)
(G,H)	(G,G)	(E,C)

Step 2 :

B	X						
C	X	X					
D	X	X	X				
E		X	X	X			
F	X	X	X		X		
G		X	X	X		X	
H	X	X	X	X	X	X	X
	A	B	C	D	E	F	G

(A,E)	(B,H)	(F,F)
(A,G)	(B,G)	(F,E)
(B,H)	(G,G)	(C,C)
(D,F)	(C,C)	(G,G)
(E,G)	(H,G)	(F,E)

Indistinguishable pairs : (A,E) , (,H) & (D,F)

Distinguishable pairs : C & G

Minimize DFA :

Step 1 :

(A,E) , (B,H) & (D,F) Indistinguishable pair C,G distinguishable pair.

Step 2 :

States in minimized DFA

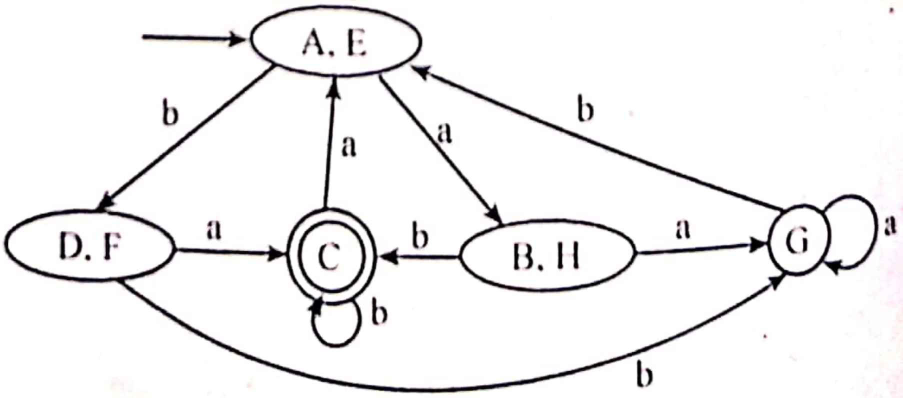
(A,E), (B,H), C; (D,F) , G

Step 3 :

δ	0	1
(A,E)	(B,H)	(D,F)
(B,H)	G	C
C	(A,E)	C
(D,F)	C	G
G	G	(A,E)

Step 4 :

(A,E) is start state & C is final state



Module - 2

3. a. **Define Regular expression and write Regular expression for the following language.**

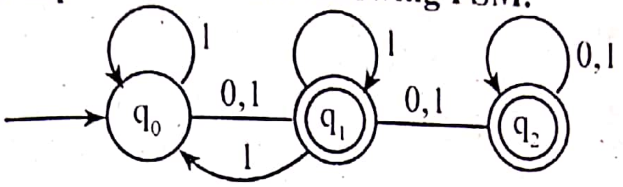
i) $L = \{a^{2n} b^{2m} \mid n \geq 0, m \geq 0\}$

ii) $L = (a^n b \mid m \geq 1, n \geq 1, nm \geq 3\}$. **(08 Marks)**

Ans. Refer Q.no. 3(a) of June/ July 2018 for definition and (ii)

i. R . E = $(aa)^* (bb)^*$

b. **Obtain the Regular expression for the following FSM.** **(08 Marks)**



Ans. Since q_2 is

R . E = 1 01* 1(0 + 1)*

OR

4. a. **Define a Regular grammar. Design regular grammars for the following languages.**

i) Strings of a's and b's with at least one a.

ii) Strings of a's and b's having strings without ending with ab.

iii) Strings of 0's and 1 's with three consecutive 0's. **(08 Marks)**

Ans. i) A grammer G is 4 - tople G = (V,T,P,S) where

V is set of variables or non - terminals

T is set of terminals

P is set of production

S is start symbol

i. V = {S,A}

T = {Q}

P = { $S \rightarrow aS$

$S \rightarrow \varepsilon$}

}

S is start symbol

ii. $S \rightarrow aA \mid bS$

$A \rightarrow aA \mid bB$

$B \rightarrow aA \mid bS \mid \varepsilon$

iii. V = {S}

T = {0,1}

P = { $S \rightarrow A\,000A$

$A \rightarrow 0A \mid 1A \mid \varepsilon$

}

S is the start symbol

b. State and prove pumping theorem for regular languages. **(08 Marks)**

Ans. Refer Q.no. 4(c) of June / July 2018.

Module-3

5. a. Define context free grammar. Design a context free grammar for the languages.
i) $L = \{0^m 1^m 2^n \mid m \geq 0, n \geq 0)$ ii) $L = \{a^i b^j \mid i \neq j, i \geq 0, j \geq 0\}$
iii) $L = \{a^n b^{n-3} \mid n \geq 3\}$. **(08 Marks)**

Ans. Refer Q.no. 5(a) of June/July 2018

i. $S \rightarrow AB$
$A \rightarrow 01 /)A1$
$B \rightarrow \varepsilon \mid 2B$

$V = \{S,A,B\}$
$T = \{0,1\}$
S is start symbol

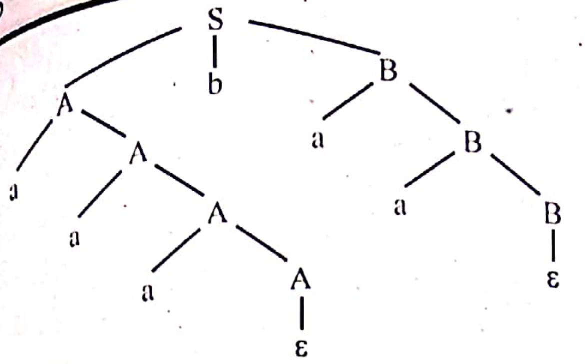
ii. $V = \{S.A.B.C\}$
$T = \{a, b\}$
$P = \{$
$S \rightarrow aSb$
$S \rightarrow A$
$S \rightarrow B$
$A \rightarrow aA \mid a$
$B \rightarrow bB \mid b$
$\}$
S is start symbol

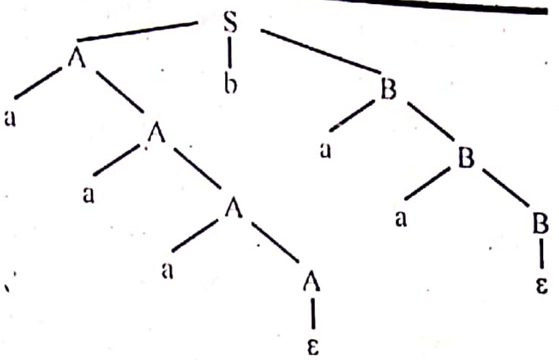
iii. $V = \{S,A\}$
$T = \{a,b\}$
$P = \{$
$S \rightarrow a\,a\,a\,A$
$A \rightarrow aA\,b \mid \varepsilon$
$\}$
S is start symbol

b. Consider the grammar G with production.
$S \rightarrow AbB$
$A \rightarrow aA \mid \in$
$B \rightarrow aB \mid bB \mid \in$
Obtain leftmost derivation , rightmost derivation and parse tree for the string aaabab. **(08 Marks)**

Ans. $S \Rightarrow A\beta B$
$\Rightarrow AbaB$
$\Rightarrow AbabB$
$\Rightarrow Abab$
$\Rightarrow aAbab$
$\Rightarrow aaAbab$
$\Rightarrow aaaAbab$
$\Rightarrow aaabab$

$\Sigma \Rightarrow A\beta B$
$\Rightarrow aAbB$
$\Rightarrow aaAbB$
$\Rightarrow aaaAbB$
$\Rightarrow aaabB$
$\Rightarrow aaabaB$
$\Rightarrow aaababB$
$\Rightarrow aaabab$





OR

6. a. **Define a PDA. Obtain a PDA to accept $L = \{a^n b^n \mid W \in \{a, b\}^*\}$. Draw the transition diagram.** **(08 Marks)**.

Ans. Refer Q.no. 6(a) of June / July 2018

$Q = \{q_0, q_7, q_2\}$

$\Sigma = \{a,b\}$

$\Gamma = \{a,Z_0\}$

$\delta : \{$

$\delta(q_0, a, z_0) = (q_0, aZ_0)$

$\delta(q_0, a, a) = (q_0, aa)$

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

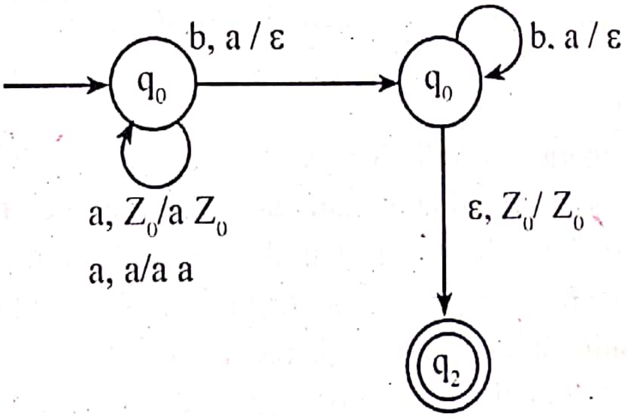
$\delta(q_1, b, a) = (q_1, \varepsilon)$

$\delta(q_1, \varepsilon, Z_0) = (q_0, Z_0)$

$q_0 \in Q$ is the start state of machine

$Z_0 \in \Gamma$ is the initial symbol on the stack

$F = \{q_2\}$ is the final state



b. **Convert the following grammar into equivalent PDA.** **(08 Marks)**

$S \rightarrow aABC$

$A \rightarrow aB|a$

$B \rightarrow bA|b$

$C \rightarrow a.$

Ans. **Step 1 :**

Push start symbol

$\delta(q_0, \varepsilon, Z_0) = (q, SZ_0)$

Step 2 :

$S \rightarrow aABC$	$\delta(q_1,a,S) = (q_1 , ABC)$
$A \rightarrow aB$	$\delta(q_1 , a , A) = (q_1 , B)$
$A \rightarrow a$	$\delta(q_1 , a , A) = (q_1 , \varepsilon)$
$B \rightarrow bA$	$\delta(q_1 , b , B) = (q_1 , A)$
$B \rightarrow b$	$\delta(q_1 , b , B) = (q_1 , \varepsilon)$
$C \rightarrow a$	$\delta(q_1 , a , c) = (q_1 , \varepsilon)$.
	$\delta(q_1 , \varepsilon , Z_0) = (q_{ff} , Z_0)$

Step 3 :

$M = (Q, \Sigma, \Gamma, \delta, q_0 , Z_0 , F)$

$Q = \{q_0 , q , q_f\}$

$\Sigma = \{a,b\}$

$\Gamma = \{S,A,B,C,Z_0\}$

δ is transition in step 2

$q_0 \in Q$ is the start symbol

$Z_0 \in \Gamma$ is initial stack symbol

$F = \{q_f\}$ is final state

Module-4

7. a. State and prove pumping lemma for context free languages. Show that $L = \{a^n b^n c^n \mid n \geq 0\}$ is not context free. **(10 Marks)**

Ans. **Statement :** Let L be the context free language and is infinite. Let Z be sufficiently long string and z L so that | z | n where n is some positive integer. If the string z can be decomposed into combination of strings z = uvwxy.

Such that |vwx| n. |vx| 1. then uv'w'y L for i=0,1,2,.......

Proof of Pumping Lemma:

By pumping lemma, it is assumed that string z L is finite and is context free language. We know that z is string of terminal which is derived by applying series of productions.

Case 1: To generate a sufficient long string z, one or more variables must be recursive. Let us assume that the language is finite, the grammar has a finite number of variables and each has finite length. The only way to derive sufficiently long string using such productions is that the grammar should have one or more recursive variables. Assume that no variable is recursive.

Since no non terminal is recursive, each variable must be defined. Since those variables are also non recursive, they to be defined in terms of terminal and other variables and so on.

From this we conclude that there is a limit length of the string that is generated from the start symbol S. this contradicts our assumption that the language is finite. Therefore, the assumption that one or more variable are non recursive is incorrect. This means that this means that one or more variable are non recursive and hence the proof.

Case 2: The string z L implies that after applying some / all production some number of times, we get finally string of terminal and the derivation stops.

Let z L is sufficiently long string and so the derivation must have involved recursive use of some non terminal A and the derivation must have the form:

Note that any derivation should start from the start symbol S.

A DFA is a 5-tuple or quintuple M=(Q, Σ, q_0, F)

Q is non-empty, finite set of states.

I is non-empty, finite state set of input alphabet.

is transition function, which is mapping from Q. X Σ to Q. for this transition function the parameters to be passed are state and input symbols. Based on the current state and input symbols, the machine may enter into another state. qo Q is the start state.

F Q is a set of accepting or final state. Note: for each input symbol a, from a given state there is exactly one transition and we are sure to which state the machine enters.

So the machine is called Deterministic Machine

Lanbncn = $\{a^n b^n c^n \mid n > 0\}$ is not a CFL

Proof. Suppose $La^n b^n c^n$ is context-free. Let p be the pumping length.

- Consider $z = a^p b^p c^p \in La^n b^n c^n$.
- Since $|z| > p$, there are u, v, w, x, y such that z = uvwxy. $|vwx| \le p$, $|vx| >0$and $uv^i wx^i y \in L$ for all i >0.
- Since $| vwx | \le p$, vwx cannot contain all three of the symbols a, b, c, because there are p bs. So vwx either does not have any as or does not have any bs or does not have any cs. Suppose, (w log) vwx does have any as. Then $uv^\circ wx^\circ y$ = uwy contains more as than either bs or cs. Hence uwy $\notin$ L.

(06 Marks)

b. **Explain Turing machine model.**

Ans. Refer Q.no. 8(b) of Dec 2017/ Jan 2018.

OR

8\. a. **Design a Turing machine to accept the language L= $\{0^n 1^n 2^n \mid n \ge 1\}$.(08 Marks)**

Ans. **Step-1:**

Replace 0 by X and move right, Go to state Q.1.

Step-2:

Replace 0 by 0 and move right, Remain on same state

Replace Y by Y and move right, Remain on same state

Replace 1 by Y and move right, go to state Q2.

Step-3:

Replace 1 by 1 and move right, Remain on same state

Replace Z by Z and move right, Remain on same state

Replace 2 by Z and move right, go to state Q3.

Step-4:

Replace 1 by 1 and move left, Remain on same state

Replace 0 by 0 and move left, Remain on same state

Replace Z by Z and move left, Remain on same state

Replace Y by Y and move left, Remain on same state

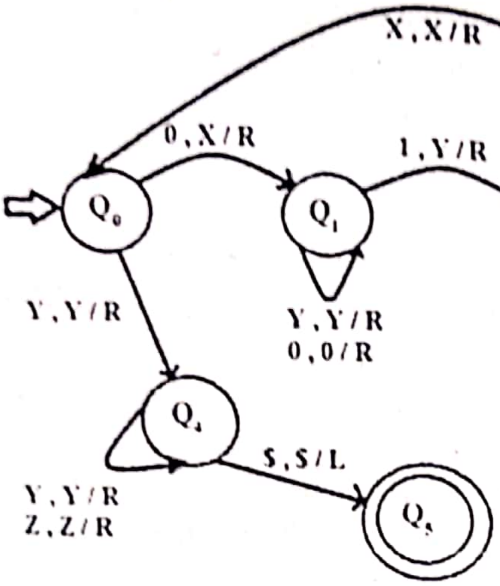
Replace X by X and move right, go to state QO.

Step-5 :

If symbol is Y replace it by Y and move right and Go to state Q4
Else go to step 1

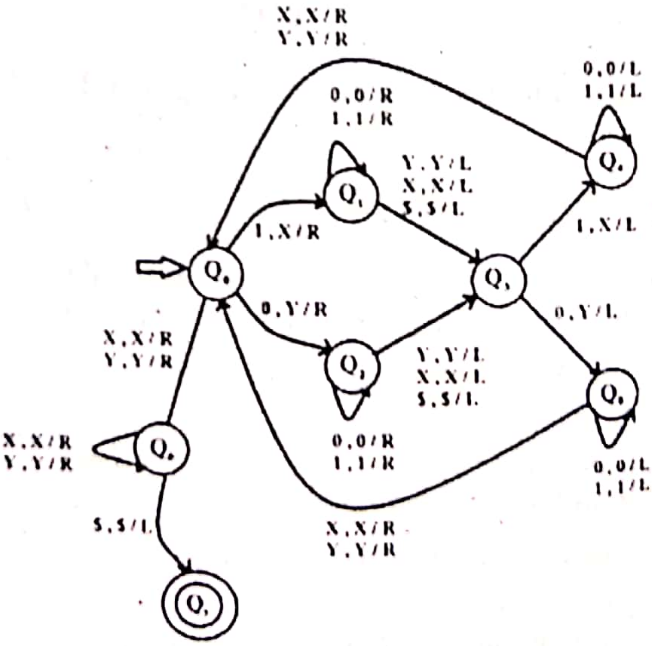
Step-6:

Replace Z by Z and move right. Remain on same state
Replace Y by Y and move right. Remain on same state
If symbol is $ replace it by $ and move left, STRING IS ACCEPTED, GO TO FINAL STATE Q5



b. Design a Turing machine to accept strings of a's and b's ending with ab or ba.
(08 Marks)

Ans.



Module-5

9. a. **Explain the following :**
i) Non deterministic Turing machine ii) Multi - tape Turing machine.(06 Marks)

Ans. REfer Q.no. 10 (a),(b) of June / July 2018.

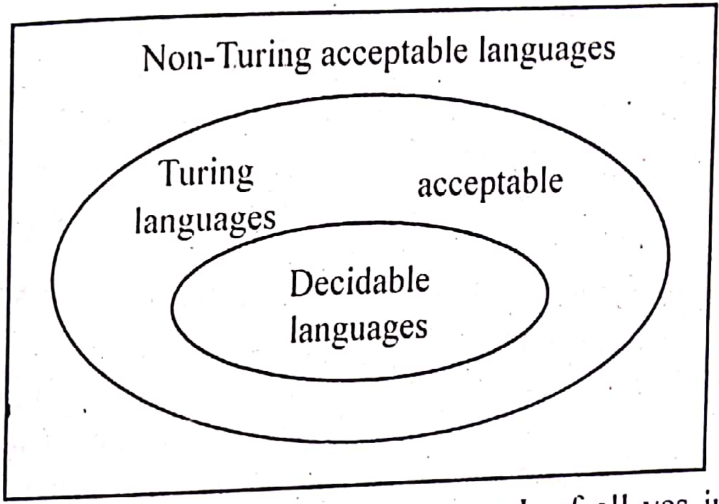
b. **Define the following :**
i) Recursively enumerable language ii) Decidable language. (06 Marks)

Ans. **Recursive Enumerable (RE) or Type -0 Language**

RE languages or type-0 languages are generated by type-0 grammars. An RE language can be accepted or recognized by Turing machine which means it will enter into final state for the strings f language and may or may not enter into rejecting state for the strings which are not part of the language. It means TM can loop forever for the strings which are not a part of the language. RE languages are also called as Turing recognizable languages.

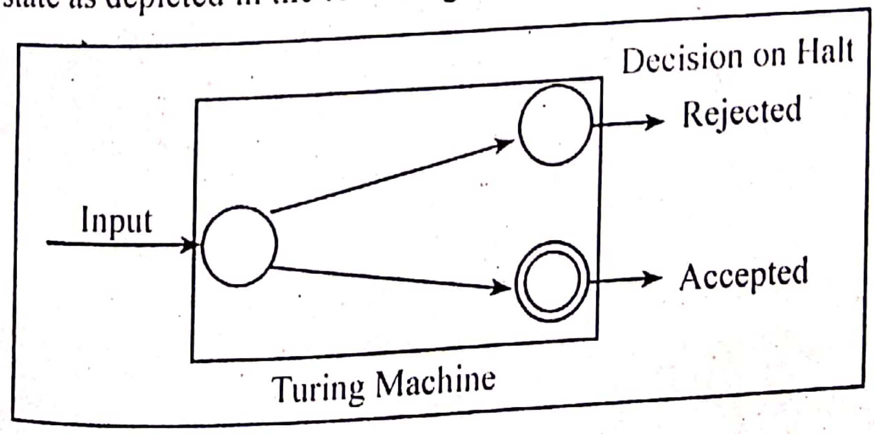
Decidable language

A language is called Decidable or Recursive if there is a Turing machine which accepts and halts on every input string w. Every decidable language is Turing-Acceptable.



A decision problem P is decidable if the language L of all yes instances to P is decidable.

For a decidable language, for each input string, the TM halts either at the accept or the reject state as depicted in the following diagram -



c. What is Post correspondence problem? (04 Marks)

Ans. Refer Q.no. 10(d) of June/ July 2018.

OR

10. a. What is Halting problem of Turing machine? (06 Marks)

Ans. Refer Q.no. 10(b) of Dec 2017 / Jan 2018.

b. Define the following : i) Quantum computer ii) Class NP. (06 Marks)

Ans. **i) Quantum Computer :** Refer Q. no 9 c of June/July 2018

ii) Class NP : The class NP consists of those problems that are verifiable in polynomial time. NP is the class of decision problems for which it is easy to check the correctness of a claimed answer, with the aid of a little extra information. Hence, we aren't asking for a way to find a solution, but only to verify that an alleged solution really is correct. Every problem in this class can be solved in exponential time using exhaustive search.

c. Explain Church Turing Thesis. (04 Marks)

Ans.

- The Church-Turing thesis concerns an effective or mechanical method in logic and mathematics.
- A method, M, is called 'effective' or 'mechanical'just in case:
- M is set out in terms of a finite number of exact instructions (each instruction being expressed by means of a finite number of symbols)
- M will, if carried out without error, always produce the desired result in a finite number of steps
- M can (in practice or in principle) be carried out by a human being unaided by any machinery except for paper and pencil
- M demands no insight or ingenuity on the part of the human being carrying it out.
- They gave an hypothesis which means proposing certain facts.
- The Church's hypothesis or Church's turing thesis can be stated as:
- The assumption that the intuitive notion of computable functions can be identified with partial recursive functions.
- This statement was first formulated by Alonzo Church in the 1930s and is usually referred to as Church's thesis, or the Church-Turing thesis.

Fifth Semester B.E. Degree Examination, CBCS - June / July 2019

Automata Theory and Compatibility

Time: 3 hrs. Max. Marks: 80

Note : Answer any FIVE full questions, selecting ONE full question from each module.

Module - 1

1. a. Define the following : i) string ii) alphabet iii) language. **(06 Marks)**

Ans. Refer Q.No. 1.a. of Dec 2018 / Jan 2019

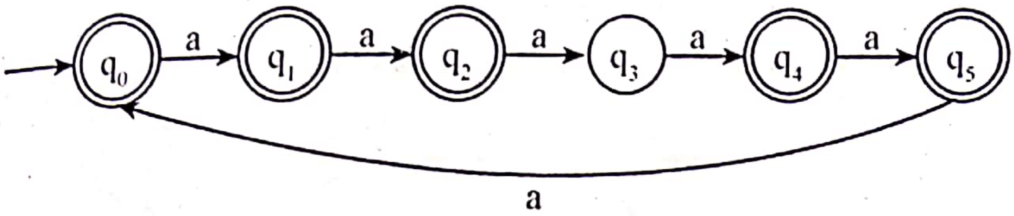
b. Design a deterministic finite State machine for the following language over Σ = {a, b}.

i) L = {W| |W| mod 3,> |W| mod 2}

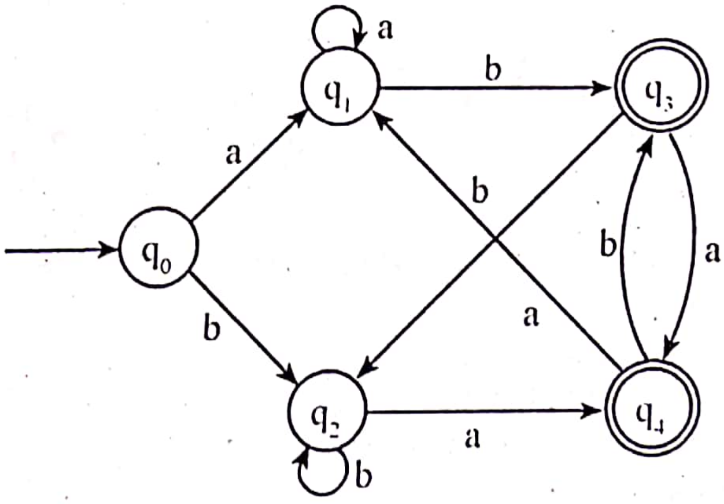
ii) L = {w | W ends either with ab or ba}. **(10 Marks)**

Ans.

i)



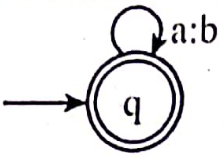
ii)



OR

a. Write a note on finite state transducers. **(07 Marks)**

ns. A finite state transducer essentially is a finite state automaton that works on two (or more) tapes. The most common way to think about transducer is as a king of "Translating machine". They read from one of the tapes and write on to the other. This for instance, is a transducer that translates aS into bS.

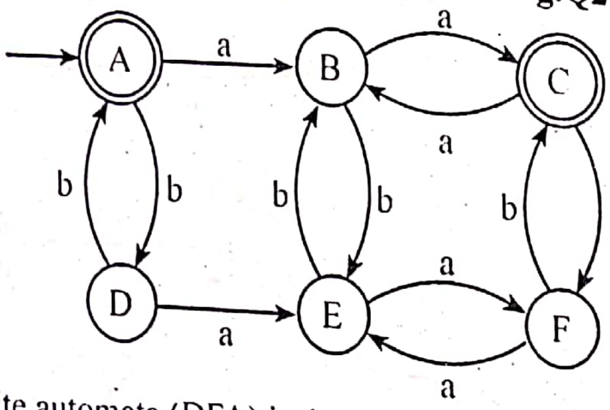


a : but the arc means that in this transition the transducer reads a from the first tape and writes b onto the second.

Transducers can however, be used in other modes that the translation made as well in the generation mode Transducers write on both tapes and in the recognition mode they read from both tapes. Further more, the direction of translation can be turned around i.e., a : b can not only be read as read a from the first tape and write b onto the second tape, but also "Read b from the second tape and write a on to the first tape. So, the above transducer behaves as follows in the different modes.

- Generation mode : it write a string of aS on one tape and a string bS on the other tape. Both strings how the same length.
- Recognition mode : it accepts when the word on the first tape consists of exactly as many aS as the word on the second tape consist of bS.
- Translation mode (left to right) : it reads aS from the first tape and writes an b for every a that it reads onto the second tape.
- Translation mode (right to left) : it reads bS from the second tape and write on a for every f that ir reads onto the first tape.

b. Define DFSM? Minimize the following FSM. [Refer Fig.Q2(b)] **(09 Marks)**



Ans. A deterministic finite automata (DFA) is described by five element tuple: $(Q, \Sigma, \delta, q_0, F)$

Q is a finite set of states

Σ is a non empty input alphabet

δ is a series of transition function

q_0 is starting state

F is final state

	a	b
→ *A	B	D
B	C	E
*C	B	F
D	E	A
E	F	B
F	E	C

Step 1 :-

	B	X				
*	C	X	X			
	D	X		X		
	E	X		X		
	F	X		X		
		A	B	C	D	E
		*		*		

Step 2 :-

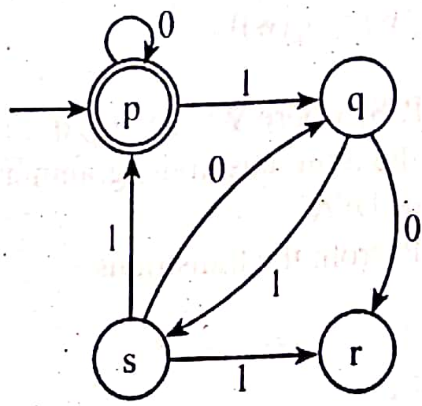
	a	b
(B, D)	(C, E)	(E, A)
(B, E)	(C, F)	(E, B)
(B, F)	(C, E)	(E, C)
(D, E)	(E, F)	(A, B)
(D, F)	(E, E)	(A, C)
(E, F)	(F, E)	(B, C)

B	X				
C	X	X			
D	X	X	X		
E	X	X	X	X	
F	X	X	X	X	X
	A	B	C	D	E

Distinguish pairs
A, B, C, D, E, F
Since there is no in distinguishable pairs
Given DFA cannot be minimized its already in minimized state.

Module-2

3. a. **Write the equivalent Regular Expression for the given Finite state machine. [Refer Fig.Q3(a)]** **(08 Marks)**



Ans. Since the problem given doesn't have any transition for 'r' for the input 'o' it cant be solved.

By assuming the transition 'r' for the input 'o' goes to 'q' we get following R.E

$= 1 * 0(11 * 0) * 0 + 1 * 0 (11 * 0) * 0 [(0 + \varepsilon) + 1(11 * 0)*0]* (0 + \varepsilon) + 1(11 * 0)*0$

b. Write the Regular Expression for the following language.

i) $\{w \in \{a, b\}^*$ with atmost one a$\}$

ii) $\{w \in \{a, b\}^*$ does not end with ba$\}$

iii) $\{w \in \{0, 1\}^*$ has substring 001$\}$

iv) $\{w \in \{0, 1\}^* \; |W|$ is even$\}$. (08 Marks)

Ans. i) $(b + a)\, b^* (\varepsilon + a)$

ii) $(a + b)^* (aa \mid ab \mid bb) |a|b| \varepsilon$

iii) $(0 + 1)^* 001 (0 + 1)^*$

iv) $((a + b)(a + b))^*$

OR

4. a. State and prove the pumping theorem for regular language. (08 Marks)

Ans. Refer Q.No. 4.c. of June / July 2018

b. Show that the language $L = \{a^n b^n \mid n > 0\}$ is not regular. (08 Marks)

Ans. **Step 1 :-** Let L is regular and n be the number of states in FA.
Consider the string $x = a^n b^n$

$$x = \overbrace{\underbrace{a\,a\,a\,a\,a\,a}_{u}\quad \underbrace{a}_{v}}^{n}\quad \overbrace{\underbrace{b\,b\,b\,b\,b\,b\,b}_{w}}^{n}$$

Step 2 :- Since $l \times l = 2n \geq n$, we can split x into uvw such that $|uv| \leq n$ and $|v| \geq 1$ as shown below.

Where $|u| = n - 1$ and $|v| = 1$ so that $|uv| = |u| + |v| = n - 1 + 1 = n$ and $|w| = n$.

According pumping lemma $uv^iw \in L$ for $i = 0, 1, 2......$

Step 3 :- if $i = 0$, v does not appear and number of a's will be less than b's which is contradict to state of pumping lemma. Hence given language is not regular.

Module-3

5. a. Define grammar. Write the CFG for the following language.

i) $L = \{w \in \{a, b\}^* \mid n_a(w) = n_b(w)\}$

ii) $L = \{a^i b^j \mid i = j + 1\}$. (08Marks)

Ans. A grammar $G = (V, T, P, S)$ where $V = \{q_0, q_1, q_2 ......\}$ is states of DFA

T is input alphabet are the terminals in the grammar

$S = q_0$ is the start state of DFA

'P' is the productions 'P' from the transitions

i) V = {S}
T = {a, b}
p = {
$S \rightarrow \varepsilon$
$S \rightarrow a\,S\,b$
$S \rightarrow b\,S\,a$
}
s is start symbol

ii) V = {S}
T = {a, b}
P = {
$S \rightarrow OA$
$A \rightarrow OA\,1$
$A \rightarrow \varepsilon$
}
s is start symbol

b. What is inherent ambiguity? Show that the language given is inherently amtriguous?
$L = \{a^n b^n c^m \mid n, m \geq 0\} \cup \{a^n b^m c^n \mid n, m \geq 0\}$ (08 Marks)

Ans. In many cases, when confronted with an ambiguous grammar 'G' it is possible to construct a new grmmar and that generates L(G) and that has less (or no) ambiguity unfortunately it is not always possible to do this. There exist context free language for which no unambiguous grammar exits. We call such languages inherently ambiguous.

$L = \{a^i b^j c^k : i, j, k \geq 0, i = j \text{ or } j = k\}$.

An alternative way to describe it is $L = \{a^n b^n c^m \mid n. m \geq 0\} \cup \{a^n b^m c^n \mid n, m \geq 0\}$ Every string in L has either (or both) the same number of a's and b's or the same number of b's and c's. L is inherently ambiguous one grammar that describes it is

$G = \{S, S_1, S_2, A, B, a, b, c\}, (a, b, c\}, R, S\}$ where
$R = \{S \rightarrow S_1 \mid S_2$
$S_1 \rightarrow S_1\,c \mid A$
$A \rightarrow a\,A\,b \mid \varepsilon$
$S_2 \rightarrow aS_2 \mid B$
$B \rightarrow b\,B\,c \mid \varepsilon\}$

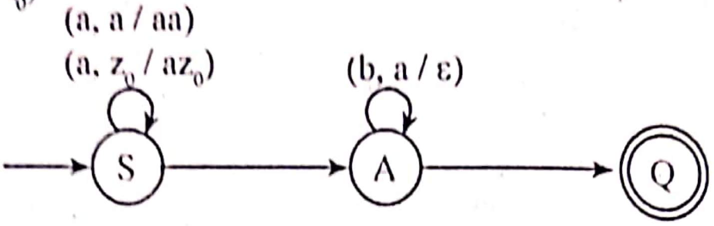
Now consider the strings in $A^n B^n C^n = \{a^n b^n c^n : n \geq 0\}$ They have two distinct derivations one through S_1 and the other through S_2 it is possible to move that L is inherently ambiguous : Given any grammar G that generates L there is at least one string with two derivation in G.

OR

6. a. Define PDA? Design PDA for the language $L = \{a^n b^m a^n \mid n, m \geq 0\}$. (06 Marks)

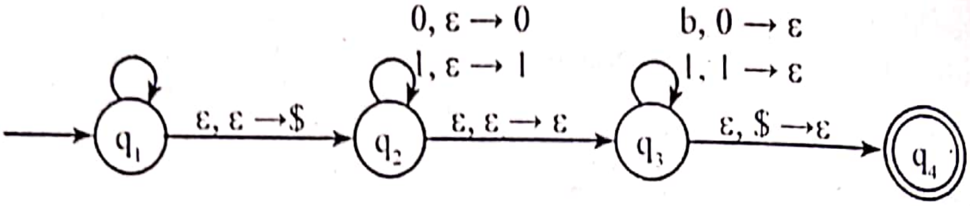
Ans. A Push Down Automata is a seven tuple
$M = (Q, \Sigma, \Gamma, \delta, q_0, z_0, F)$
where Q is set of finite states
Σ is set of input alphabets
Γ is set of stack alphabets
δ - transition form $Q\ (\Sigma \cup \varepsilon) \times \Gamma$ to finite Suv set of $Q \times P^*$
$q_0 \in Q$ is the start state
$z_0 \in P$ is the initial symbol on the stack
$F \leq Q$ is set of final states

$\delta(S, a, z_0) = (A, a\, z_0)$
$\delta(S, a, a) = (S, a\, a)$
$\delta(A, b, a) = A$
$\delta(A, a, az_0) = (A, \varepsilon)$
$\delta(A, \varepsilon, z_0) = (Q, z_0)$



b. Convert the following language from CFG to PDA L = {ww^R | w ∈ {0, 1 }*}. **(06 Marks)**

Ans.



c. Convert the following CFG to CNF E → E + E | E * E | (E) | id. **(04 Marks)**

Ans. E → E E' | (E)
E' → +E | *E
E → io |
V = {E}
T = {id, + E}
P = {
E → EE' | (E)
E' → +E | *E
E → id
}

Module-4

7. a. Prove that the language L = {a^n b^n c^n | n ≥ 0} is not context free. **(08 Marks)**

Ans. Suppose this language is contact free; then it has a context free grammar. Let K be the constant associated with this grammar by the pumping Lemma. Consider the string a^{K}b^{K}c^K, which is in L and has length greater than K.

By the pumping Lemma this must be representable as uvxyz, such that all uv^{i}xy^{i}z are also in L. This is impossible, since

- either v and y cannot contain a mixture of letters from {a, b, c}; otherwise they would be in the wrong order for uv^{2}xy^{2}z
- if v or y contain just 'a's, 'b's or 'c's, then uv^{2}xy^{2}z cannot maintain the balance between the three letters (it can, of course maintain the balance between two) QεD

b. **Prove that CFL are not closed under intersection, complement or difference?** **(08 Marks)**

Ans. Refer Q.No. 7.a. of Dec 2017 / Jan 2018

OR

8. a. **Design a Turing machine to accept L = {a^n b^n c^n | n ≥ 0}.** **(08 Marks)**

Ans. Refer Q.No. 8.b. of Dec 2017 / Jan 2018

b. **Define a turning machine. Explain the working of a turning machine.(05 Marks)**

Ans. Refer Q.No. 8.a. of Dec 2017 / Jan 2018

c. **Write a note on multitape machine.** **(03 Marks)**

Ans. Refer Q.No. 9.a. of Dec 2017 / Jan 2018

Module-5

9. **Write a short notes on :**

a. **Growth rate of function** **(05 Marks)**

b. **Church-turning thesis** **(06 Marks)**

c. **Linear bounded automata.** **(05 Marks)**

Ans. a. **Growth rate of function :** One of the most important problems in computer science is to get the best measure of the growth rates of algorithms, best being those algorithms whose run times grow the slowest as a function of the size of their input. Efficiency can mean survival of a company. For example, a sort of measure 0(2 n) on a database of millions of customers may take several days to run, whereas one of measure 0(n · log n) may take only a few minutes!

However, the big O estimate, does not necessarily give the best measure of the growth rate of a function. One can say that the growth rate of a sequential search is O (n^2). but one knows the number of comparisons is approximately proportional to n, n the number of input items. We would like to say that sequential search is O(n) (it is), but the notion of big O is not precise enough. Therefore, in this section we define theta Θ notation to more precisely measure the growth rate of functions and big omega Ω notation. The most important of these is O . We also define o and ω notation.

b. **Church-turning thesis :** Refer Q.No. 10.c. of Dec 2018 / Jan 2019

c. **Linear bounded automata :** Refer Q.No. 10.c. of June / July 2019

OR

10. **Write a short notes on :**

a. **Post correspondence problem** **(05 Marks)**

b. **Halting problem in turning machine** **(05 Marks)**

c. **Various types of turning machine.** **(06 Marks)**

Ans. a. **Post correspondence problem :** Refer Q.No. 10.c. of Dec 2017 / Jan 2018

b. **Halting problem in turning machine :** Refer Q.No. 10.b. of Dec 2017 / Jan 2018

c. Various types of turning machine :

1) Multiple track
2) Shift over Turing Machine
3) Nondeterministic
4) Two way Turing Machine
5) Multitape Turing Machine
6) Multidimensional Turing Machine
7) Composite Turing Machine
8) Universal Turing Machine

Refer Q. no 9.a. of Dec 2018/Jan 2019