Central Concepts of Automata theory.

$+$ = { One or more occurance } ⟶ Σ^+, $a^+ = \{a, aa \ldots$

$*$ = { zero or more occurance } ⟶ Σ^*, $a^* = \{\epsilon, a, aa, \ldots$

Relation between Σ^+ & Σ^*

$\Sigma^* = \Sigma^+ \cup \{\epsilon\}$.

$= \Sigma^+ \cup \Sigma^0$

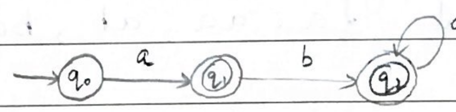
Finite Automata :-

1.) Deterministic FA (DFA)
2) Non - Deterministic FA (NFA)
3) Epsilon NFA (E - NFA).

} Design

Notations (common to DFA, NFA & E-NFA) $(Q, \Sigma, \delta, q_0, F)$.

1) Q : set of states.
$\{q_0, q_1, q_2\}$.



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

$q_0 = \{q_0\}$ = initial state

F = set of final states $= \{q_1, q_2\}$.

δ - transition function.

$\delta(\text{state}_i, \text{alphabet}) = \text{state}_j$

$\delta(q_0, a) = q_1$

$\delta(q_1, b) = q_2$

$\delta(q_2, a) = \delta(q_2, b) = q_2$.

Central Concepts of Automata Theory :-

1) Alphabets (Σ) $\Sigma = \{a, b\}$.
2) Strings
3) Languages
4) Powers (Σ^k) $\Sigma^2 = \{aa, ab, ba, bb\}$

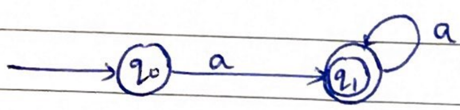
*) In DFA for each state all the elements should be defined.

I. Deterministic FA (DFA)

1) Design a DFA to accept strings of a's having atleast one a $(Q, \Sigma, \delta, q_0, F)$

Sol^n →
$L = \{a, aa, aaa \ldots \ldots a^n\}$.

$\Sigma = \{a\}$. $Q = \{q_0, q_1\}$. $q_0 = \{q_0\}$ $F = \{q_1\}$

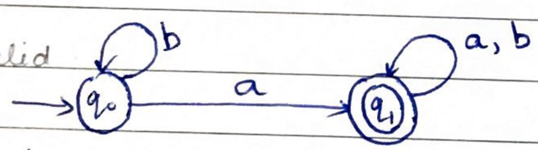


$\delta(q_0, a) = \{q_1\}$
$\delta(q_1, a) = \{q_1\}$.

2) Design a DFA to accept strings of a's & b's having atleast one a.

Sol^n →
$L = \{a, aa, ab, ba \ldots\ldots\}$

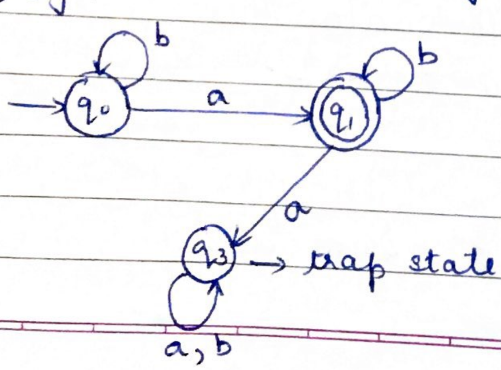
b - non final stage → Not valid
a → final stage → valid
aa → FS → V
ab → FS → V
bb - NFS → NV



$\Sigma = \{a, b\}$.
$Q = \{q_0, q_1\}$.
$q_0 = \{q_0\}$
$F = \{q_1\}$.
$\delta(q_0, a) = \{q_1\}$
$\delta(q_0, b) = \{q_0\}$
$\delta(q_1, a) = \{q_1\}$
$\delta(q_1, b) = \{q_1\}$.

3) a's & b's having exactly one 'a'

Sol^n →
$\Sigma = \{a, b\}$.

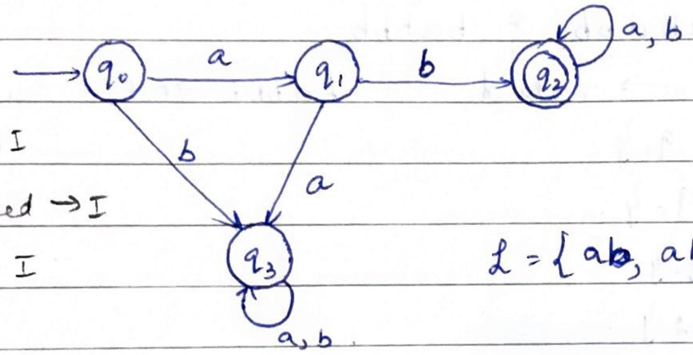


$Q = \{q_0, q_1, q_3\}$.
$q_0 = \{q_0\}$.
$F = \{q_1\}$.
$\delta(q_0, a) = \{q_1\}$
$\delta(q_0, b) = \{q_0\}$

$\delta(q_1, a) = \{q_3\}$.
$\delta(q_1, b) = \{q_1\}$.
$\delta(q_3, a) = \{q_3\}$
$\delta(q_3, b) = \{q_3\}$.
$L = \{a, ab, ba, abb \ldots\}$.

4) a's & b's, starting with ab.

Sol^n → $\Sigma = \{a, b\}$



a → NFS → I

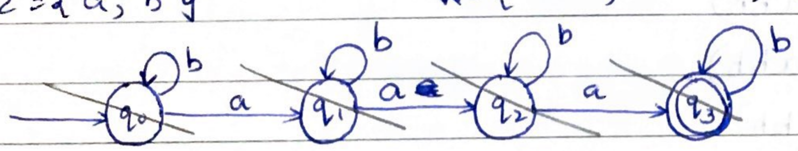
b → not allowed → I

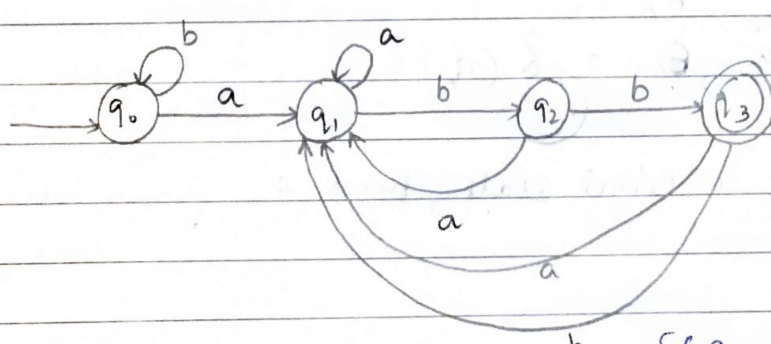
a a → T → I

$L = \{ab, abab, abbb \ldots\}$.

$Q = \{q_0, q_1, q_2, q_3\}$.
$q_0 = \{q_0\}$ $F = \{q_2\}$
$\delta(q_0, a) = \{q_1\}$. $\delta(q_0, b) = \{q_3\}$ $\delta(q_1, a) = \{q_3\}$.
$\delta(q_1, b) = \{q_2\}$ $\delta(q_2, a) = \{q_2\}$ $\delta(q_2, b) = \{q_2\}$.
$\delta(q_3, a) = \{q_3\}$ $\delta(q_3, b) = \{q_3\}$.

5) a's & b's ending with abb.
Sol^n → $\Sigma = \{a, b\}$ $L = \{abb, aabb, babb \ldots\}$.



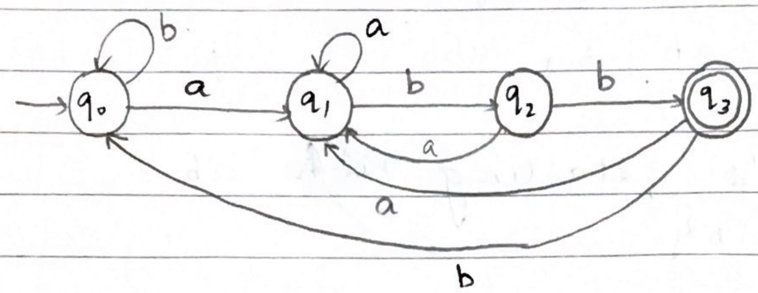


$q_0 = \{q_0\}$.
$F = \{q_3\}$.
$\delta(q_0, a) = \{q_1\}$.
$\delta(q_0, b) = \{q_0\}$
$\delta(q_1, a) = \{q_1\}$
$\delta(q_1, b) = \{q_2\}$
$\delta(q_2, a) = \{q_1\}$.
$\delta(q_2, b) = \{q_3\}$.
$\delta(q_3, a) = \{q_1\}$ $\delta(q_3, b) = \{q_1\}$

Design DFA which accepts the following languages.

1) Strings of a's & b's ending with abb. Check the i/p strings ababb & babbb.

Sol^n ⇒



Tracing for ababb & babbb

ababb → FS → valid   babbb → NFS → invalid.

$\delta(q_0, a) = \{q_1\}$.

$\delta(q_0, b) = \{q_0\}$.

$\delta(q_1, a) = \{q_1\}$

$\delta(q_1, b) = \{q_2\}$.

$\delta(q_2, a)$

$\delta(q_2; b)$

$\delta(q_3, a)$

$\delta(q_3, b)$

δ → Transi^n Func^n (TF)

$\hat{\delta}$ → Extended TF

(i) ababb

$\hat{\delta}(q_0, ababb)$

$= \hat{\delta}(q_1, babb)$

$= \hat{\delta}(q_2, abb)$

$= \hat{\delta}(q_1, bb)$

$= \hat{\delta}(q_2, b)$

$= \hat{\delta}(q_3, \epsilon) = \cancel{\hat{\delta}(q}$

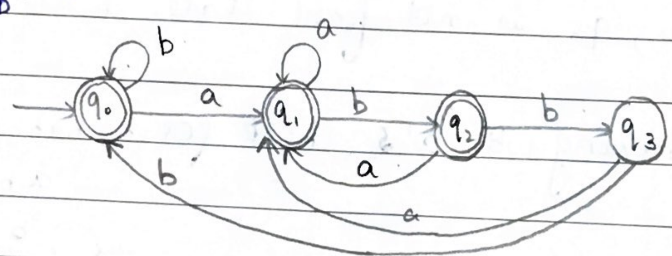
q_3 is final state, hence the given string is valid.

(ii) babbb

$\hat{\delta}(q_0, babbb)$
$= \hat{\delta}(q_0, abbb)$
$= \hat{\delta}(q_1, bbb)$
$= \hat{\delta}(q_2, bb)$
$= \hat{\delta}(q_3, b)$
$= \hat{\delta}(q_0, \epsilon)$

q_0 is not the final state, hence given string is invalid.

2) Strings of a's & b's not ending with abb. check for ababb, babbb



$F = \{q_0, q_1, q_2\}$ $q_0 = \{q_0\}$.

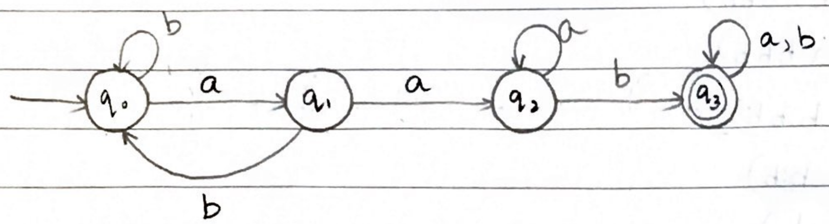
(i) $\hat{\delta}(q_0, ababb)$
$= \hat{\delta}(q_1, babb)$
$= \hat{\delta}(q_2, abb)$
$= \hat{\delta}(q_1, bb)$
$= \hat{\delta}(q_2, b)$
$= \hat{\delta}(q_3, \epsilon)$, q_3 is not the final state, hence invalid.

(ii) babbb

$\hat{\delta}(q_0, babbb)$
$= \hat{\delta}(q_0, abbb)$
$= \hat{\delta}(q_1, bbb)$
$= \hat{\delta}(q_2, bb)$

Page No.:
Date:

3. a's & b's having substring aab

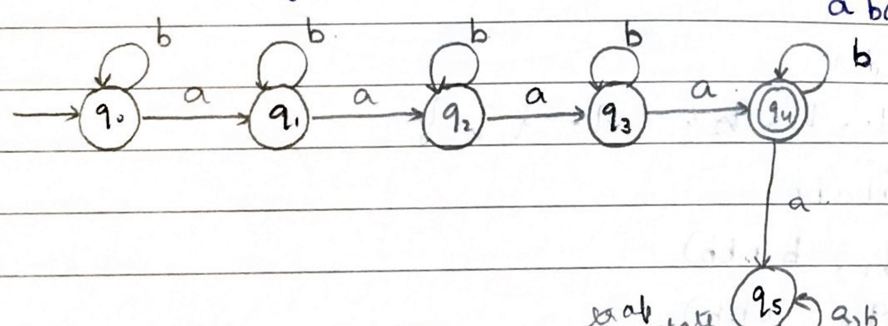


1) ababb

$F = \{q_3\}$, $q_0 = \{q_0\}$.
$\Sigma = \{a, b\}$. $S = \{q_0, q_1, q_2, q_3\}$.

$\hat{\delta}(q_0, ababb)$
$= \hat{\delta}(q_1, babb)$
$= \hat{\delta}(q_0, abb)$
$= \hat{\delta}(q_1, bb)$
$= \hat{\delta}(q_0, b)$
$= \hat{\delta}(q_0, \epsilon)$, q_0 is not final state, hence invalid

4) a's & b's having 4 a's, check for babab, abaaab, ababaaa.

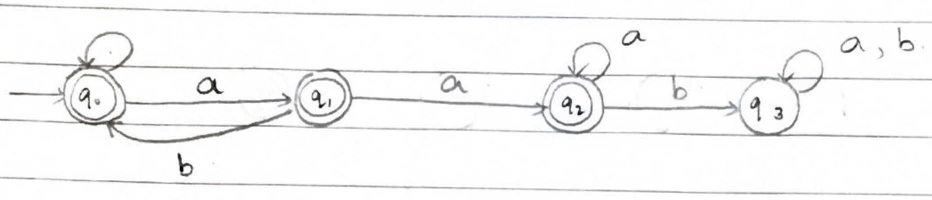


1.) babab, $\hat{\delta}(q_0, babab) = \hat{\delta}(q_0, abab) = \hat{\delta}(q_1, bab) = \delta(q_1, ab)$
$= \hat{\delta}(q_2, b) = \hat{\delta}(q_2, \epsilon)$ $\rightarrow$ invalid ($\because q_2$ is not FS)

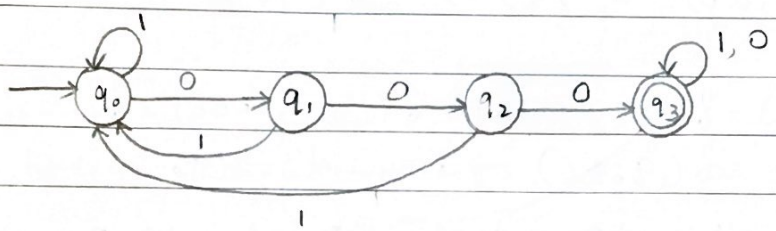
2) abaaab ; $\hat{\delta}(q_0, abaaab) = \hat{\delta}(q_1, baaab) = \hat{\delta}(q_1, aaab)$
$= \hat{\delta}(q_2, aab) = \hat{\delta}(q_3, ab) = \hat{\delta}(q_4, b) = \hat{\delta}(q_4, \epsilon)$
$\Rightarrow$ valid ($\because q_4$ is ~~valid~~ ~~not~~ FS)

3.) ababaaa ; $\hat{\delta}(q_0, ababaaa) = \hat{\delta}(q_1, babaaa) = \hat{\delta}(q_1, abaaa)$
$= \hat{\delta}(q_2, baaa) = \hat{\delta}(q_2, aaa) = \hat{\delta}(q_3, aa)$
$= \hat{\delta}(q_4, a) = \hat{\delta}(q_5, \epsilon)$ $\longrightarrow$ invalid ($\because q_5$ is not FS

5) a's & b's except those having the substring aab



6) strings of 0's & 1's having 3 consecutive 0's (substring '000'). i/p string : (10001 & 010101)



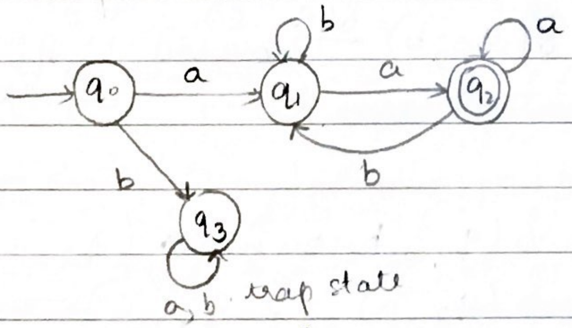
(i) w : 10001

$\hat{\delta}(q_0, 10001) = \hat{\delta}(q_0, 0001) = \hat{\delta}(q_1, 001) = \hat{\delta}(q_2, 01)$
$= \hat{\delta}(q_3, 1) = \hat{\delta}(q_3, \epsilon) \rightarrow$ valid ($\because$ q_3 is FS)

(ii) w : 010101

$\hat{\delta}(q_0, 010101) = \hat{\delta}(q_1, 10101) = \hat{\delta}(q_0, 0101) = \hat{\delta}(q_1, 101)$
$= \hat{\delta}(q_0, 01) = \hat{\delta}(q_1, 1) = \hat{\delta}(q_0, \epsilon) \rightarrow$ invalid ($\because$ q_0 is not FS)

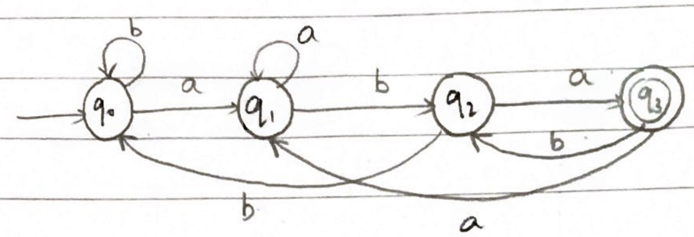
7) $L = \{awa \mid w \in (a+b)^n, n \geq 0\}$, i/p strings ababa & aabbb.



(i) ababa, $\hat{\delta}(q_0, ababa) = \hat{\delta}(q_1, baba) = \hat{\delta}(q_1, aba) = \hat{\delta}(q_2, ba)$
$= \hat{\delta}(q_1, a) = \hat{\delta}(q_2, \epsilon) \rightarrow$ valid ($\because$ q_2 is FS)

(ii) babbb, $\hat{\delta}(q_0, babbb) = \hat{\delta}(q_4, abbb) = \hat{\delta}(q_4, bbb) = \hat{\delta}(q_4, bb)$
$= \hat{\delta}(q_4, b) = \hat{\delta}(q_4, \epsilon) \rightarrow$ invalid ($\because$ q_4 is not FS)

8) $L = \{waba \mid w \in \{a,b\}^*\}$ i/p : ababab & babbb



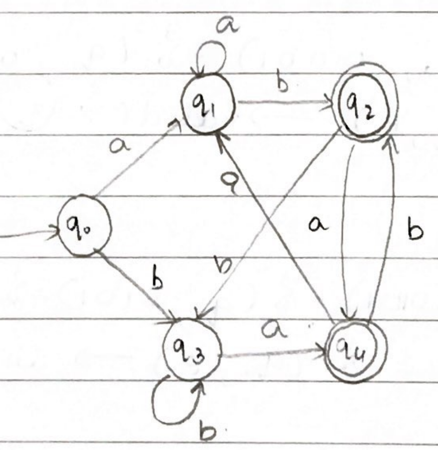
(i) w : ababab

$\hat{\delta}(q_0, ababab) = \hat{\delta}(q_1, babab) = \hat{\delta}(q_2, abab) = \hat{\delta}(q_3, bab)$
$= \hat{\delta}(q_2, ab) = \hat{\delta}(q_3, b) = \hat{\delta}(q_2, \epsilon) \longrightarrow$ invalid ($\because$ q_2 is not FS

(ii) babbb

$\hat{\delta}(q_0, babbb) = \hat{\delta}(q_0, abbb) = \hat{\delta}(q_1, bbb) = \hat{\delta}(q_2, bb) = \hat{\delta}(q_0, b)$
$= \hat{\delta}(q_0, \epsilon) \rightarrow$ invalid $\rightarrow$ (q_0 is not FS)

9) Strings of a's & b's ending with ab or ba i/p : ababb. ababa & babab.



1) w : ababb

$\hat{\delta}(q_0, ababb) = \hat{\delta}(q_1, babb) = \hat{\delta}(q_2, abb) = \hat{\delta}(q_4, bb)$
$= \hat{\delta}(q_2, b) = \hat{\delta}(q_3, \epsilon) \rightarrow$ invalid ($\because$ q_3 is not FS)

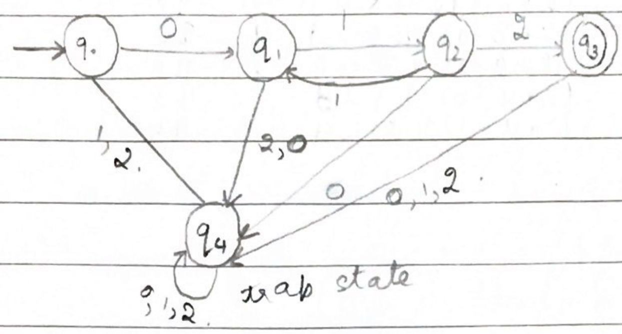
2) w : ababa

$\hat{\delta}(q_0, ababa) = \hat{\delta}(q_1, baba) = \hat{\delta}(q_2, aba) = \hat{\delta}(q_4, ba)$
$= \hat{\delta}(q_2, a) = \hat{\delta}(q_4, \epsilon) \longrightarrow$ valid ($\because$ q_4 is FS)

3) w : babab

$\hat{\delta}(q_0, babab) = \hat{\delta}(q_3, abab) = \hat{\delta}(q_4, bab) = \hat{\delta}(q_2, ab)$
$= \hat{\delta}(q_4, b) = \hat{\delta}(q_2, \epsilon) \rightarrow$ valid ($\because$ q_2 is FS)

10) Strings of a's & b's 0, 1 & 2 begining with a 0 followed by odd numbers of only 1's & ending with 2



(i) 01112

$\hat{\delta}(q_0, 01112) = \hat{\delta}(q_1, 1112) = \hat{\delta}(q_2, 112) = \hat{\delta}(q_1, 12)$
$= \hat{\delta}(q_2, 2) = \hat{\delta}(q_3, \epsilon) \rightarrow$ valid ($\because q_3$ is FS)

(ii) 01122

$\hat{\delta}(q_0, 01122) = \hat{\delta}(q_1, 1122) = \hat{\delta}(q_2, 122) = \hat{\delta}(q_1, 22)$
$= \hat{\delta}(q_4, 2) = \hat{\delta}(q_4, \epsilon) \rightarrow$ invalid ($\because q_4$ is not FS)

Divisible by k problems:-

$\delta(q_i, d) = q_j$, where $j = (r * i + d) \bmod k$

i :- remainder after dividing by k.

k : ~~divid~~ divisor.

r : radix

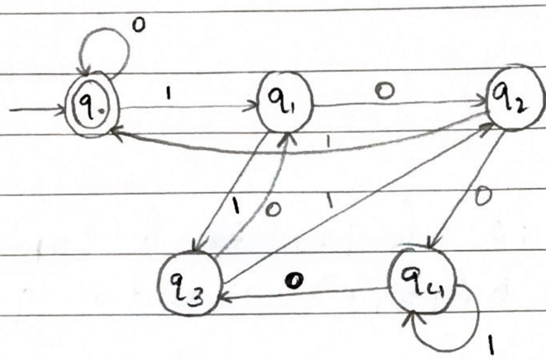
d : inputs.

Construct a DFA which accepts strings of 0's & 1's where the value of each string is represented as binary number & only the strings representing zero module five should be accepted.

eg : 0000, 0101, 1010, 1111

Soln → $i = (0, 1, 2, 3, 4)$, $k : 5$, $r : 2$, $d : (0, 1)$

i	d	j = (r*i + d) mod k	qj
q_0 (0)	0	j = (2*0+0) % 5 = 0	q_0
	1	j = (2*0+1) % 5 = 1	q_1
q_1 (1)	0	(2*1+0) % 5 = 2	q_2
	1	(2*1+1) % 5 = 3	q_3
q_2 (2)	0	(2*2+0) % 5 = 4	q_4
	1	(2*2+1) % 5 = 0	q_0
q_3 (3)	0	(2*3+0) % 5 = 1	q_1
	1	(2*3+1) % 5 = 2	q_2
q_4 (4)	0	(2*4+0) % 5 = 3	q_3
	1	(2*4+1) % 5 = 4	q_4



eg :- 1 = 1 ✗   1010 = 10 ✓

11 = 3 ✗   1011 = 11 ✗

111 = 7 ✗

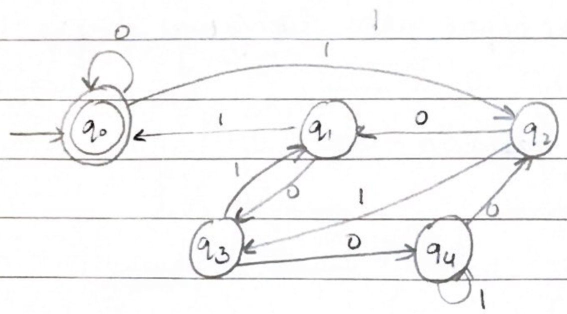
1111 = 15 ✓

Q) Obtain DFA that accepts set of all strings when the interpreted in reverse as a binary integer is divisible by 5

Solⁿ → Eg :- 0, 10011, 1001100, 0101

i = 0, 1, 2, 3, 4   k : 5,   r : 2   d : (0, 1)

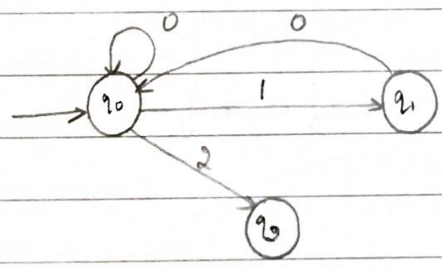
i	d	j = (r*i + d) mod k
0	0	
	1	
1	0	
	1	
2	0	
	1	
3	0	
	1	
4	0	
	1	



Construct a DFA to accept decimal string divisible by 3.

$i : 0, 1, 2$ $k : 3,$ $r = 10$, $d : 0, 1, 2, 3, 4, 5, 6, 7, 8, 9$

i	d	$j = (r * i + d) \bmod k$	q_j
~~4~~ 0	(0,3,6 9)	0	q_0
	(1,4,7)	1	q_1
	(2,5,8)	2	q_2
1	(0,3,6,9)	1	q_1
	(1,4,7)	2	q_2
~~1~~	(2,5,8)	0	q_0
2	(0,3,6,9)	2	q_2
	(1,4 7)	0	q_0
	(2,5,8)	1	q_1



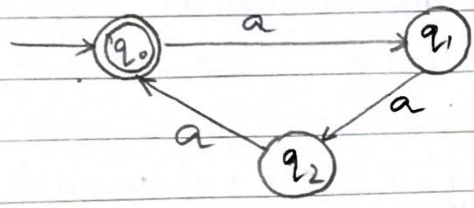
Page No.:
Date:

Q) Design the DFA which accepts the language $L = \{w : |w| \bmod 3$
$\Sigma = \{a\}$.

→ $L = \{\epsilon, aaa, aaaaaa, aaaaaa \ldots\}$.

① $Q = \{q_0, q_1, q_2\}$.

② Transition function (δ)



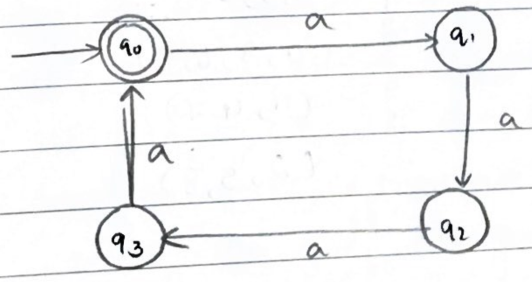
Q) $L = \{w : |w| \bmod 4 = 0\}$. $\Sigma = \{a\}$.

→

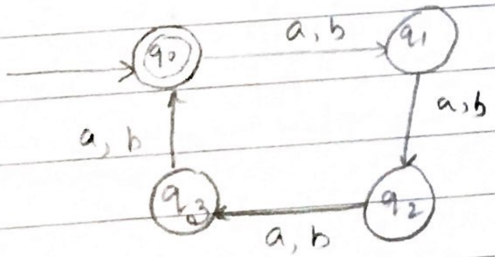
① $Q = \{q_0, q_1, q_2, q_3\}$.

② TF

$\delta(q_0, a) = q_1$
$\delta(q_1, a) = q_2$
$\delta(q_2, a) = q_3$
$\delta(q_3, a) = q_0$



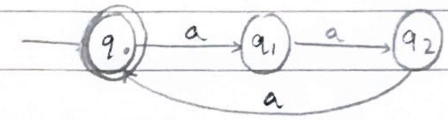
Q) $L = \{w : |w| \bmod 4 = 0\}$. $\Sigma = \{a, b\}$



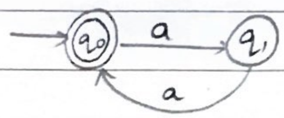
(only length matters not the orders in these particular problems)

Q) $L = \{ w : |w| \bmod 3 \geq |w| \bmod 2 \}$ $\quad \Sigma = \{a\}$

① D_1 : $|w| \bmod 3$ $\quad Q_1 = \{q_0, q_1, q_2\}$



② D_2 : $|w| \bmod 2$. $\quad Q_2 = \{q_0, q_1\}$



③ $Q = Q_1 \times Q_2$

$= \{q_0, q_1, q_2\} \times \{q_0, q_1\}$.

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

④ TF.

$\delta((q_0, q_0), a) = \cancel{\phi}\ \cancel{q_1 \cup} \{q_1, q_1\}$.

$\delta((q_0, q_1), a) = \{q_1, q_2\}$

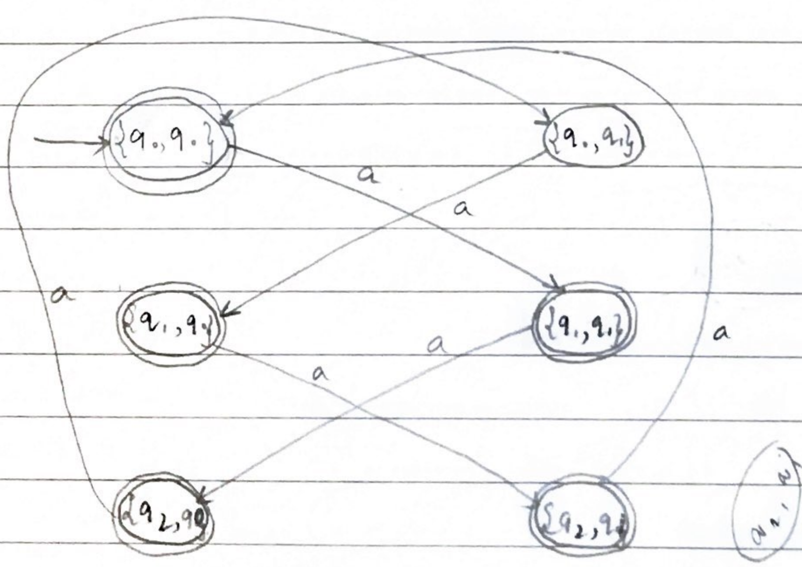
$\delta((q_1, q_0), a) = \{q_2, q_1\}$.

$\delta((q_1, q_1), a) = \{q_2, q_0\}$.

$\delta((q_2, q_0), a) = \{q_0, q_1\}$.

$\delta((q_2, q_1), a) = \{q_0, q_0\}$.

⑤

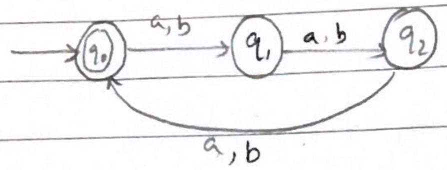


Q) $L = \{w : |w| \bmod 3 = 0 \ \& \ |w| \bmod 2 = 0\}$ $\Sigma = \{a, b\}$.

check for $w = bbbaab$

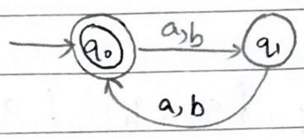
Soln → I> D_1 $Q_1 = \{q_0, q_1, q_2\}$.

$\{|w| \bmod 3 = 0\}$.



II> D_2 $Q_2 = \{q_0, q_1\}$.

$\{|w| \bmod 2 = 0\}$



III>

$Q = Q_1 \times Q_2$

$Q = \{(q_0,q_0)\ (q_0,q_1)\ (q_1,q_0)\ (q_1,q_1)\ (q_2,q_0)\ (q_2,q_1)\}$.

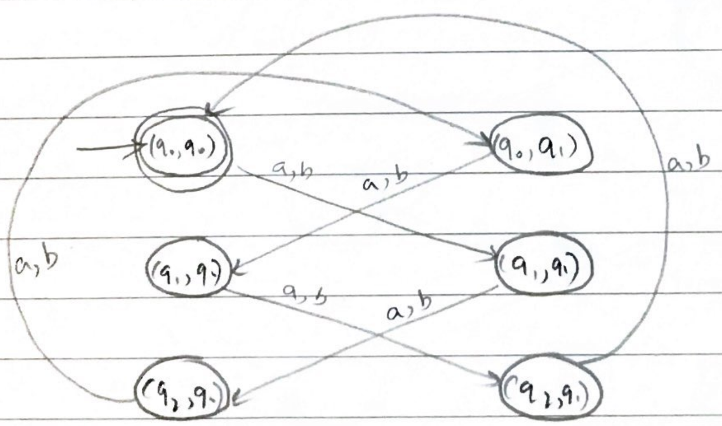
IV> $\delta((q_0,q_0), a) = (q_1,q_1)$

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

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

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

V>



Page No.:
Date:

$w = bbbaab$

$\hat{\delta}(\{q_0, q_0\}, bbbaab)$

$= \hat{\delta}(\{q_1, q_1\}, bbaab)$

$= \hat{\delta}(\{q_2, q_0\}, baab)$

$= \hat{\delta}(\{q_0, q_1\}, aab)$

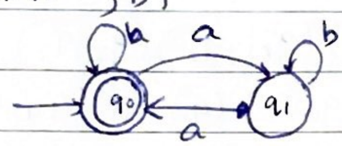
$= \hat{\delta}(\{q_1, q_0\}, ab)$

$= \hat{\delta}(\{q_2, q_1\}, b)$

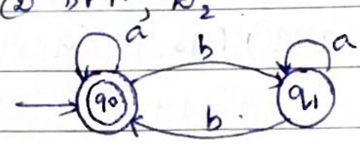
$= \hat{\delta}(\{q_0, q_0\}, \epsilon\} \rightarrow FS \rightarrow$ valid

Q) $N_a(w) \bmod 2 = 0$ & $N_b(w) \bmod 2 = 0$

① DFA, D_1



② DFA, D_2



$Q_1 = \{q_0, q_1\}$

$Q_2 = \{q_0, q_1\}$

③ $Q = Q_1 \times Q_2$

$= \{(q_0, q_0), (q_0, q_1), (q_1, q_0), (q_1, q_1)\}$

Transiⁿ Funcⁿs

④ $\delta((q_0, q_0), a) = \{q_1, q_0\}$

$\delta((q_0, q_0), b) = \{q_0, q_1\}$

$\delta((q_0, q_1), a) = \{q_1, q_1\}$

$\delta((q_0, q_1), b) = \{q_0, q_0\}$

$\delta((q_1, q_0), a) = \{q_0, q_0\}$

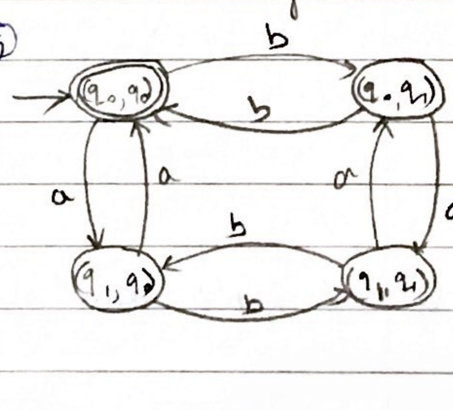
$\delta((q_1, q_0), b) = \{q_1, q_1\}$

$\delta((q_1, q_1), a) = \{q_0, q_1\}$

$\delta((q_1, q_1), b) = \{q_1, q_0\}$

Transiⁿ diagram

⑤



δ	a	b
$\rightarrow * \{q_0, q_0\}$	$\{q_1, q_0\}$	$\{q_0, q_1\}$
$\{q_0, q_1\}$	$\{q_1, q_1\}$	$\{q_0, q_0\}$
$\{q_1, q_0\}$	$\{q_0, q_0\}$	$\{q_1, q_1\}$
$\{q_1, q_1\}$	$\{q_0, q_1\}$	$\{q_1, q_0\}$

Transiⁿ table

(*) → final state

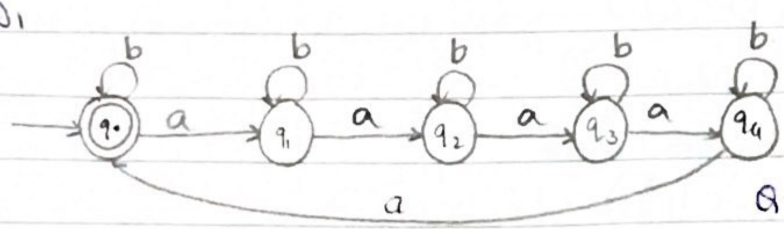
(→) → initial state

done in tutorial

Q) $N_a(w) \bmod 5 = 0$

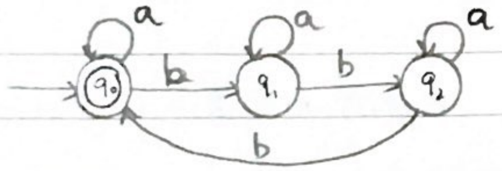
$N_b(w) \bmod 3 = 0$

① D_1



$Q = \{q_0, q_1, q_2, q_3, q_4\}$

② D_2



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

③ $Q = Q_1 \times Q_2$

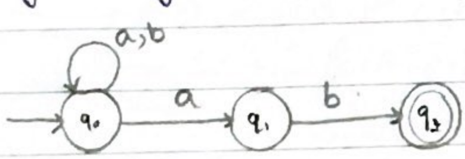
$= \{(q_0, q_0)(q_0, q_1)(q_0, q_2)(q_1, q_0)(q_1, q_1)(q_1, q_2)(q_2, q_0)(q_2, q_1)$
$(q_2, q_2)(q_3, q_0)(q_3, q_1)(q_3, q_2)(q_4, q_0)(q_4, q_1)(q_4, q_2)\}$

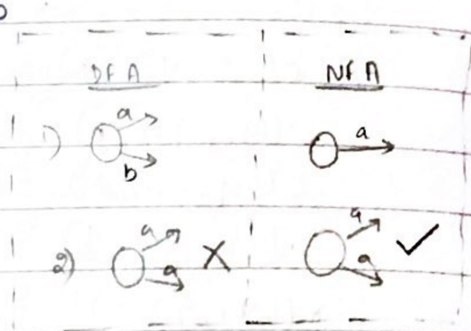
④ TF :

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

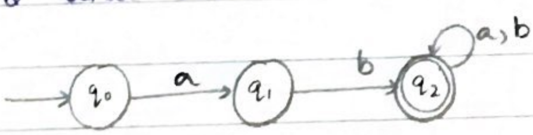
II Non-Deterministic FA (NFA)

* Q) Design an NFA strings of a's & b's such that each and every strings ends with ab





Q) a's & b's and starts with ab

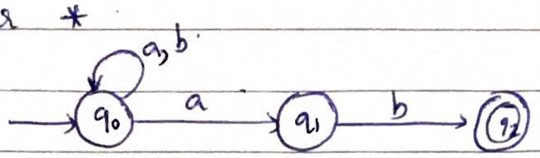


$w = abab$

$\hat{\delta}(q_0, abab) = \hat{\delta}(q_1, bab) = \hat{\delta}(q_2, ab) = \hat{\delta}(q_2, b) = \hat{\delta}(q_2, \epsilon)$

$= q_2$ (final state ∴ valid)

for *



$w = abab$

$\hat{\delta}(q_0, abab)$

$= \hat{\delta}(\{q_0, q_1\}, bab)$.

$= \hat{\delta}(\{q_0, q_2\}, ab)$. $\longrightarrow \{q_0, q_1\} \cup \phi$

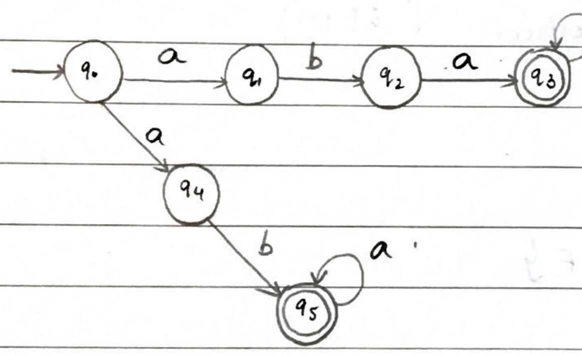
$= \hat{\delta}(\{q_0, q_1\}, b)$

$= \hat{\delta}(\{q_0, q_2\}, \epsilon)$.

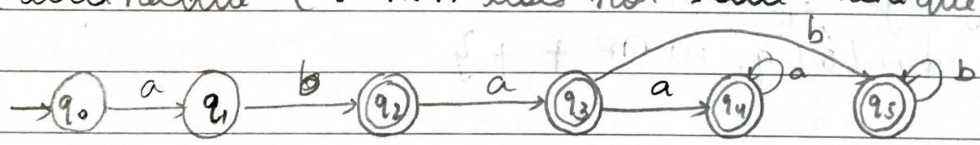
$= \{q_0, q_2\}$.

Q.) Design an NFA which accepts the following language.

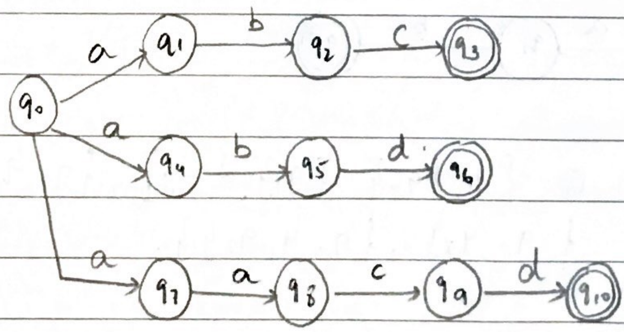
$L = \{ w \mid w \in abab^n \text{ or } aba^n, n \geq 0 \}$



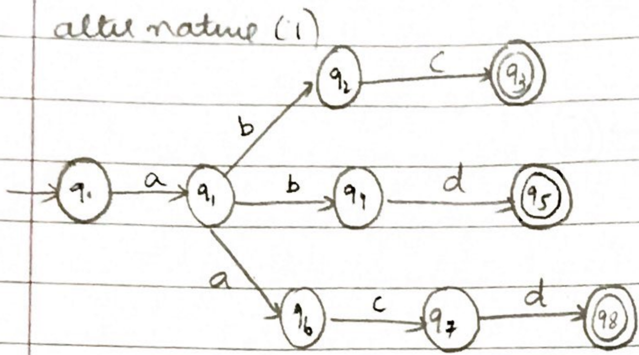
(alternative (∵ NFA does not have unique soln))



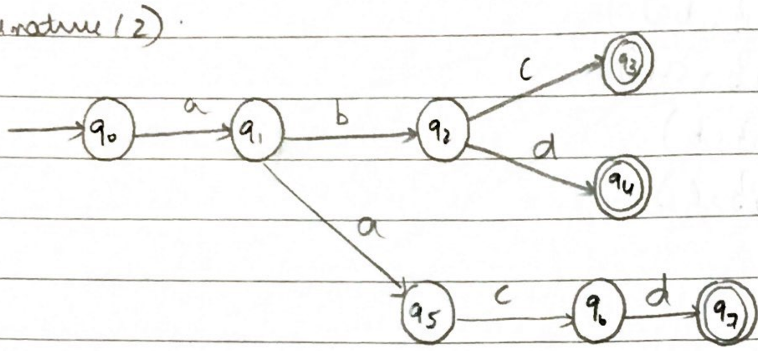
Q) abc, abd & aacd (accepts strings of this form).



alternative (1)



alternative (2) :



Conversion from NFA to DFA

① Subset construc^n method (SCM)

② Lazy Evalua^n method (LEM)

Language of DFA :-

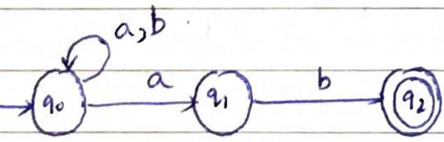
$$L = \{ w \mid \hat{\delta}(q_0, w) \in F \}.$$

Language of NFA

$$L = \{ w \mid \hat{\delta}(q_0, w) \cap F \neq \phi \}$$

Convert the foll NFA to its equivalent DFA using SCM

(Q.)



sol^n →

Subsets of $Q = \{ \phi, \{q_0\}, \{q_1\}, \{q_2\}, \{q_0, q_1\}, \{q_1, q_2\}, \{q_0, q_2\}, \{q_0, q_1, q_2\} \}$.

$2^3 = 8$ states

Page No.:
Date:

a) for state ϕ :-

$\delta(\phi, a) = \phi$ $\qquad$ $\delta(\phi, b) = \phi$.

(b) for states $\{q_0\}$.

$\delta(\{q_0\}, a) = \{q_0, q_1\}$ $\qquad$ $\delta(\{q_0\}, b) = \{q_0\}$.

(c) for states $\{q_1\}$.

$\delta(\{q_1\}, a) = \phi$ $\qquad$ $\delta(\{q_1\}, b) = \{q_2\}$

(d) for state $\{q_2\}$.

$\delta(\{q_2\}, a) = \phi$ $\qquad$ $\delta(\{q_2\}, b) = \phi$.

e) $\{q_0, q_1\}$.

$\delta(\{q_0, q_1\}, a) = \{q_0, a\} \cup \{q_1, a\}$.
$= \{q_0, q_1\} \cup \phi$
$= \{q_0, q_1\}$

$\delta(\{q_0, q_1\}, b) = \{q_0, b\} \cup \{q_1, b\}$
$= \{q_0\} \cup \{q_2\}$
$= \{q_0, q_2\}$

f) $\{q_0, q_2\}$.

$\delta(\{q_0, q_2\}, a) = \{q_0, a\} \cup \{q_2, a\}$.
$= \{q_0, q_1\} \cup \phi$
$= \{q_0, q_1\}$.

$\delta(\{q_0, q_2\}, b) = \{q_0, b\} \cup \{q_2, b\}$
$= \{q_0\} \cup \cancel{\{q_0\}}\ \phi$.
$= \{q_0\}$.

g) $\{q_1, q_2\}$.

$\delta(\{q_1, q_2\}, a) = \{q_1, a\} \cup \{q_2, a\}$
$= \phi \cup \phi$
$= \phi$

$\delta(\{q_1, q_2\}, b) = \{q_1, b\} \cup \{q_2, b\}$
$= \{q_2\} \cup \phi$
$= \{q_2\}$.

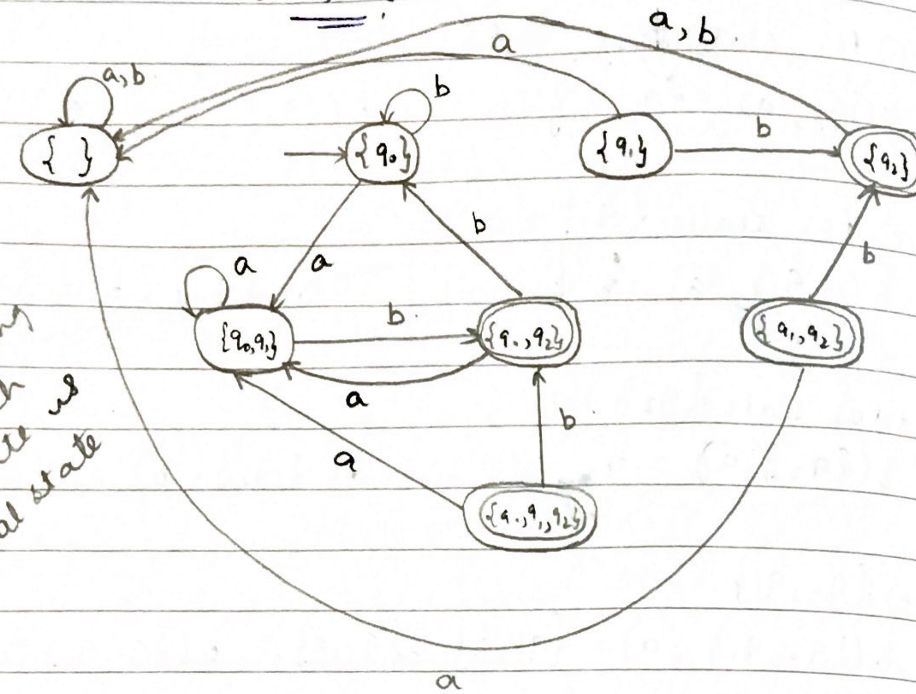
h) $\{q_0, q_1, q_2\}$.

$\delta(\{q_0, q_1, q_2\}, a) = \{q_0, a\} \cup \{q_1, a\} \cup \{q_2, a\}$.
$= \{q_0, q_1\} \cup \phi \cup \phi$
$= \{q_0, q_1\}$.

$\delta(\{q_0, q_1, q_2\}, b) = \{q_0, b\} \cup \{q_1, b\} \cup \{q_2, b\}$

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

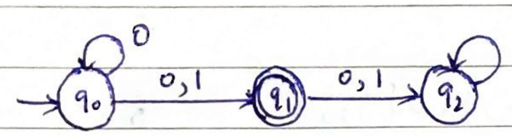
$= \{q_0, q_2\}$



which ever state in NFA is marked final, all the corresponding states in DFA which contains that state is marked as final state

Q) Convert the following NFA to its equivalent DFA

~~Subset~~



max of one occurence of 1

Subset of $Q = \{\phi, \{q_0\}, \{q_1\}, \{q_2\}, \{q_0, q_1\}, \{q_0, q_2\}, \{q_1, q_2\}, \{q_0, q_1, q_2\}\}$

a) ϕ

$\delta(\phi, 0) = \phi \qquad \delta(\phi, 1) = \phi$

b) $\{q_0\}$

$\delta(\{q_0\}, 0) = \{q_0, q_1\} \qquad \delta(\{q_0\}, 1) = \{q_1\}$

c) $\{q_1\}$

$\delta(\{q_1\}, 0) = \{q_2\} \qquad \delta(\{q_1\}, 1) = \{q_2\}$

a) $\{q_2\}$

$\delta(\{q_2\}, 0) = \{\} \qquad \delta(\{q_2\}, 1) = \{q_2\}$

e) $\{q_0, q_1\}$

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

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

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

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

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

$= \{q_1, q_2\}$.

f) $\{q_0, q_2\}$.

$\delta(\{q_0, q_2\}, 0) = \{q_0, 0\} \cup \{q_2, 0\}$.

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

$= \{q_0, q_1\}$.

$\delta(\{q_0, q_2\}, 1) = \{q_0, 1\} \cup \{q_2, 1\}$.

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

$= \{q_1, q_2\}$.

g) $\{q_1, q_2\}$.

$\delta(\{q_1, q_2\}, 0) = \{q_1, 0\} \cup \{q_2, 0\}$.

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

$= \{q_2\}$.

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

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

$= \{q_2\}$.

h) $\{q_0, q_1, q_2\}$.

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

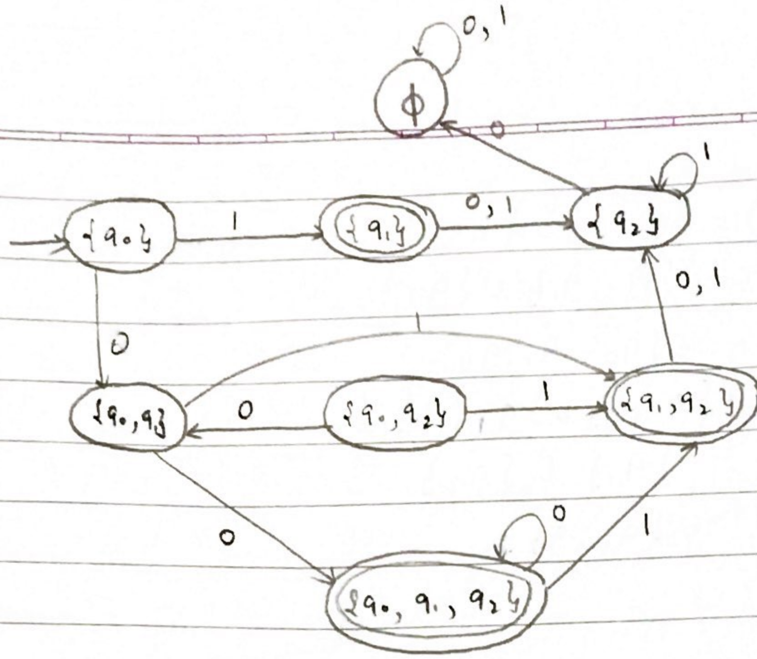
$= \{q_0, q_1\} \cup \{q_2\} \cup \{\}$

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

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

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

$= \{q_1, q_2\}$



W = 0001

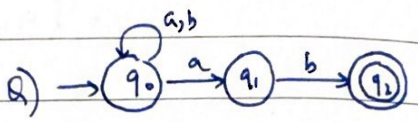
$\hat{\delta}(\{q_0\}, 0001) = \hat{\delta}(\{q_0, q_1\}, 001) = \hat{\delta}(\{q_0, q_1, q_2\}, 01)$

$= \hat{\delta}(\{q_0, q_1, q_2\}, 1) = \hat{\delta}(\{q_1, q_2\}, \epsilon)$

valid ($\because$ FS)

NFA to DFA conversion:

II> Lazy Evaluation method :- Q)



Sol^n →

① state $\{q_0\}$.

$\delta(\{q_0\}, a) = \{q_0, q_1\}$.

$\delta(\{q_0\}, b) = \{q_0\}$.

② State $\{q_0, q_1\}$

$\delta(\{q_0, q_1\}, a) = \delta(\{q_0\}, a) \cup \delta(\{q_1\}, b) = \{q_0, q_1\} \cup \phi$.

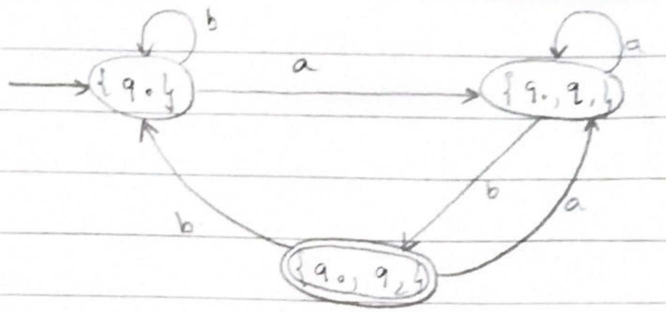
$= \{q_0, q_1\}$.

$\delta(\{q_0, q_1\}, b) = \{q_0, q_2\}$

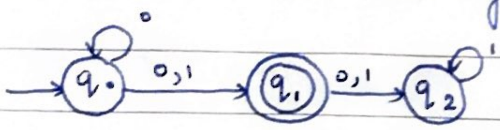
③ $\{q_0, q_2\}$

$\delta(\{q_0, q_2\}, a) = \{q_0, q_1\}$.

$\delta(\{q_0, q_2\}, b) = \{q_0\}$



Q.) Convert NFA to DFA using lazy evaluaⁿ method :-



solⁿ →

① State $\{q_0\}$.

$\delta(\{q_0\}, 0) = \{q_0, q_1\}$.

$\delta(\{q_0\}, 1) = \{q_1\}$.

② State $\{q_0, q_1\}$.

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

$\delta(\{q_0, q_1\}, 1) = \{q_1, q_2\}$.

③ State $\{q_1\}$.

$\delta(\{q_1\}, 0) = \{q_2\}$.

$\delta(\{q_1\}, 1) = \{q_2\}$.

④ State $\{q_0, q_1, q_2\}$

$\delta(\{q_0, q_1, q_2\}, 0) = \cancel{\{q_1, q_2\}} \{q_0, q_1, q_2\}$.

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

⑤ State $\{q_1, q_2\}$.

$\delta(\{q_1, q_2\}, 0) = \{q_2\}$

$\delta(\{q_1, q_2\}, 1) = \{q_2\}$

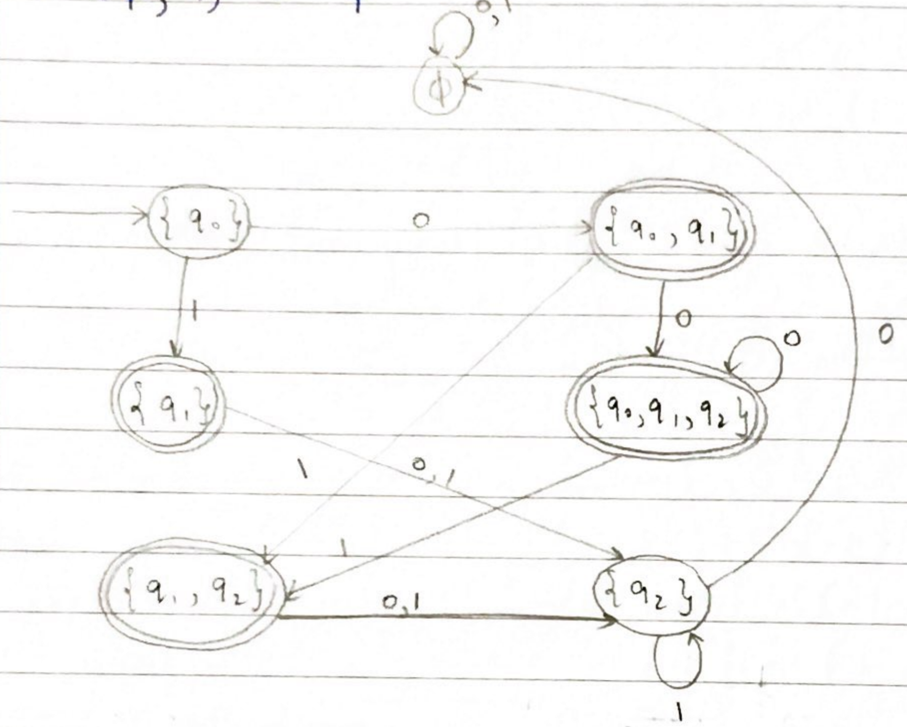
⑥ State $\{q_2\}$

$\delta(\{q_2\}, 0) = \phi$ $\delta(\{q_2\}, 1) = \{q_2\}$.

State ϕ.

$\delta(\phi, 0) = \phi$.

$\delta(\phi, 1) = \phi$.



① $w = 000$

$\hat{\delta}(\{q_0\}, 000)$

$\hat{\delta}(\{q_0, q_1\}, 00)$

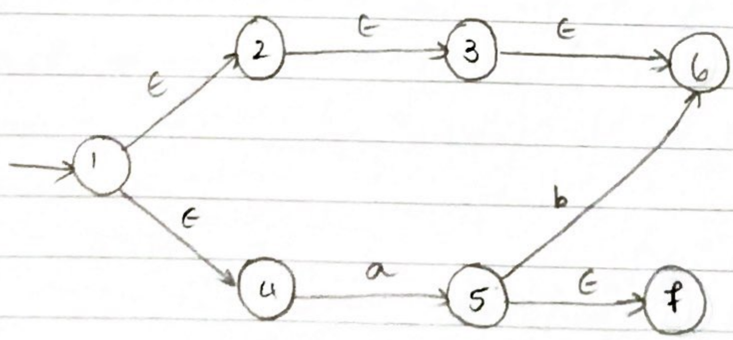
$\hat{\delta}(\{q_0, q_1, q_2\}, 0)$

$\hat{\delta}(\{q_0, q_1, q_2\}, \epsilon)$

$= \{q_0, q_1, q_2\}$. valid ($\because$ FS)

III NFA with ϵ-transitions (ϵ-NFA)

① eg:-

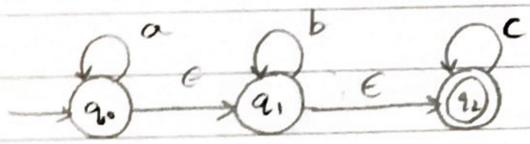


E-CLOSURE ( ) (E - CLOSE ( ))

E - closure (q) = { }

1) E - closure (1) = {1, 2, 3, 4, 6}
2) E - closure (2) = {2, 3, 6}
3) E - closure (3) = {3, 6}
4) E - closure (4) = {4}
5) E - closure (5) = {5, 7}
6) E - closure (6) = {6}
7) E - closure (7) = {7}

Q) Convert the following E-NFA to its equivalent DFA.



$[a^n b^n c^0]$

Sol^n →

a) E - closure (q_0) = $\{q_0, q_1, q_2\}$
E - closure (q_1) = $\{q_1, q_2\}$
E - closure (q_2) = $\{q_2\}$

b) State $\{q_0, q_1, q_2\}$
$\delta(\{q_0, q_1, q_2\}, a) = \{\{q_0\} \cup \phi \cup \phi\} = \{q_0\}$ = E-closure (q_0)
$\delta(\{q_0, q_1, q_2\}, b) = \{\phi \cup \{q_1\} \cup \phi\} = \{q_1\}$ = E-closure (q_1)
$\delta(\{q_0, q_1, q_2\}, c) = \phi \cup \phi \cup \{q_2\} = \{q_2\}$ = E-closure (q_2)
E-closure $(q_0) = \{q_0, q_1, q_2\} = \delta(\{q_0, q_1, q_2\}, a)$
$\delta(\{q_0, q_1, q_2\}, b) = \{q_1, q_2\}$
$\delta(\{q_0, q_1, q_2\}, c) = \{q_2\}$

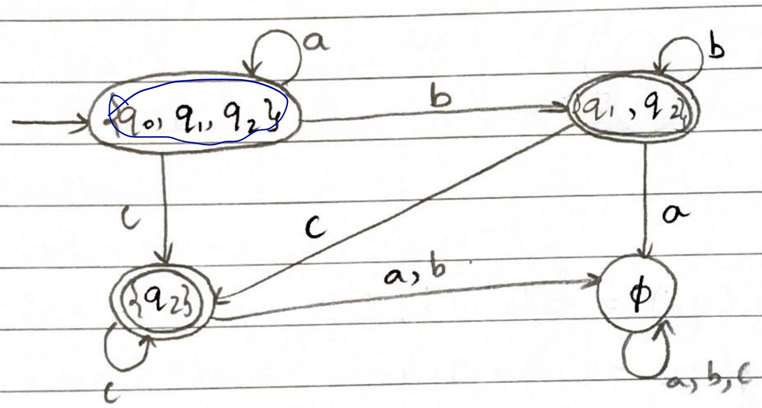
c) State $\{q_1, q_2\}$
$\delta(\{q_1, q_2\}, a) = \phi \cup \phi$ = E - closure (ϕ) = ϕ
$\delta(\{q_1, q_2\}, b) = q_1 \cup \phi$ = E - closure (q_1)
= $\{q_1, q_2\}$

Page No.:
Date:

$\delta(\{q_1, q_2\}, c) = \phi \cup q_2 = \epsilon\text{-Closure}(q_2)$
$= \{q_2\}$

d) State $\{q_2\}$.
$\delta(\{q_2\}, a) = \phi = \epsilon\text{-closure}(\phi) = \phi$
$\delta(\{q_2\}, b) = \phi = \epsilon\text{-Closure}(\phi) = \phi$
$\delta(\{q_2\}, c) = q_2 = \epsilon\text{-Closure}(q_2) = \{q_2\}$.

e) State ϕ.
$\delta(\phi, a) = \phi = \epsilon\text{-Closure}(\phi) = \phi$
$\delta(\phi, b) = \phi = \epsilon\text{-closure} = \phi$
$\delta(\phi, c) = \phi = \epsilon\text{-Closure} = \phi$



$w = abc$
$= \delta(\{q_0, q_1, q_2\}, abc)$
$= \delta(\{q_0, q_1, q_2\}, bc)$
$= \delta(\{q_1, q_2\}, c)$
$= \delta(\{q_2\}, \epsilon)$
$= \{q_2\}$ since $\{q_2\}$ is final state $\rightarrow$ valid.

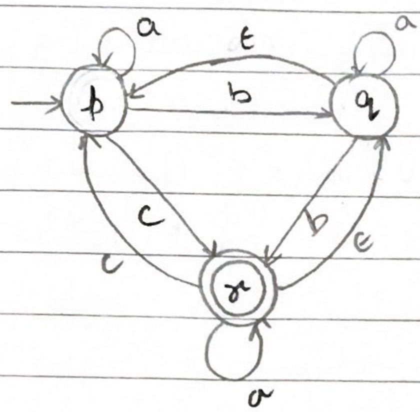
Q) Consider the following ϵ-NFA.

δ	ϵ	a	b	c
$\rightarrow p$	ϕ	$\{p\}$	$\{q\}$	$\{r\}$
q	$\{p\}$	$\{q\}$	$\{r\}$	ϕ
$* r$	$\{q\}$	$\{r\}$	ϕ	$\{p\}$

ϕ in table means not defined

① compute ϵ-closure of each state.
② convert to DFA.

Sol^n,



a) ϵ-closure $(\cancel{q}\,p) = \{p\}$.
ϵ-closure $(q) = \{p, q\}$
ϵ-closure $(r) = \{r, q, p\}$.

b) i) State $\{p\}$.
$\delta(\{p\}, a) = \{p\} = \epsilon\text{-closure}(p) = \{p\}$
$\delta(\{p\}, b) = \{q\} = \epsilon\text{-closure}(p, q) = \{p, q\}$.
$\delta(\{p\}, c) = \{r\} = \epsilon\text{-closure}(\cancel{p, q}\ r) = \{p, q, r\}$.

~~(ii) State $\{p, q\}$.~~
~~$\delta(\{p, q\}, a) = \{p\} \cup \{q\} = \{p, q\}$.~~
~~$\delta(\{p, q\}, b$~~

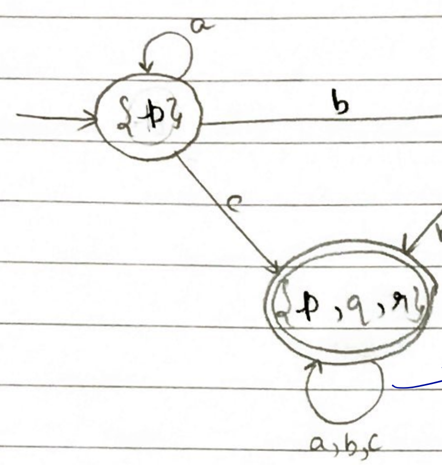
(ii) State $\{p, q\}$
$\delta(\{p, q\}, a) = \{p\} \cup \{q\} = \epsilon\text{-closure}(p, q)$
$= \epsilon\text{-closure}(p) \cup \epsilon\text{-closure}(q)$
$= \{p\} \cup \{p, q\} = \{p\,q\}$.
$\delta(\{p, q\}, b) = \{q\} \cup \{r\} = \epsilon\text{-closure}(q, r)$
$= \epsilon\text{-closure}(q) \cup \epsilon\text{-closure}(r)$
$= \{p, q\} \cup \{p, q, r\} = \{p, q, r\}$.
$\delta(\{p, q\}, c) = \{r\} \cup \{\} = \epsilon\text{-closure}\{r\} = \{p, q, r\}$.

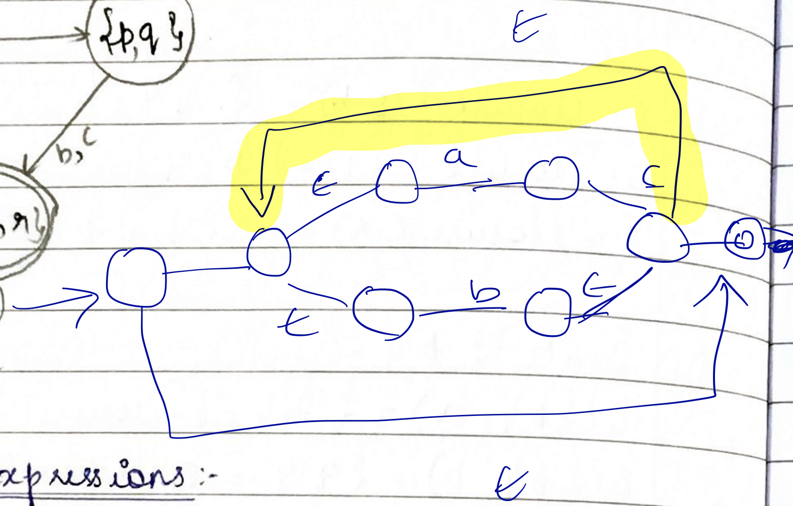
(iii) State $\{p, q, r\}$

$\delta(\{p,q,r\}, a) = \{p\} \cup \{q\} \cup \{r\} = \epsilon\text{-closure}(p,q,r)$
$= \{p,q,r\}$

$\delta(\{p,q,r\}, b) = \{q\} \cup \{r\} \cup \{\} = \epsilon\text{-closure}(q,r)$
$= \{p,q,r\}$

$\delta(\{p,q,r\}, c) = \{r\} \cup \{\} \cup \{p\} = \epsilon\text{-closure}(p,r)$
$= \{p,q,r\}$





Unit - 2 : Regular Expressions :-

Q) Write the R.E for the following language.

1) Set of ~~set~~ strings of a's & b's of any length including empty string. - $(a+b)^*$
2) a's & b's ending with abb. - $(a+b)^* abb$.
3) a's & b's starting with ab. - $ab(a+b)^*$.
4) a's & b's having substring aa - $(a+b)^* aa (a+b)^*$
5) String consisting of any no. of a's, b's, ~~c's~~, c's - $a^* b^* c^*$
6) String consisting of atleast one a ~~followed by any number of b's & c's~~ followed by atleast one b & atleast one c - $aa^* bb^* cc^*$.
7) a's & b's ending either with a or bb - $(a+b)^*(a+bb)$
8) Strings consisting of even number of a's ~~odd~~ followed by odd no. of b's - $(aa)^* b (bb)^* \equiv (aa)^* (bb)^* b$
9) Strings of a's & b's having length 2. - $(aa+ab+ba+bb) \equiv (a+b)(a+b)$

10) Strings of a's & b's with length ≤ 2.
$- \epsilon + a + b + (a+b)(a+b)$

or

$(a+b+\epsilon)(a+b+\epsilon)$

11.) Strings of a's & b's having even length $- ((a+b)(a+b))^* \equiv (aa+ab+ba+bb)^*$

12) Strings of a's & b's having odd length $- ((a+b)(a+b))^*(a+b)$

$\equiv$

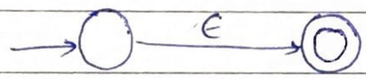
$(aa+ab+ba+bb)^*(a+b)$

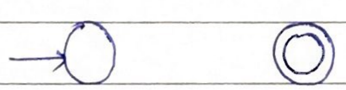
13) Strings of a's & b's with alternate a's & b's $-$ ~~(ab)*(ba)*~~
~~$-(ab)^* + (ba)^* + (ab)^*a + (ba)^*b$~~
$\rightarrow (\epsilon+a)(ba)^*(\epsilon+b)$

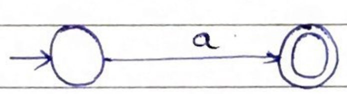
~~14)~~

Regular Expression and Finite Automata:

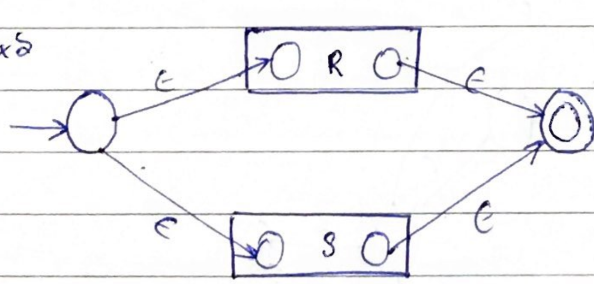
I RE to FA (ϵ-NFA)

①  ϵ

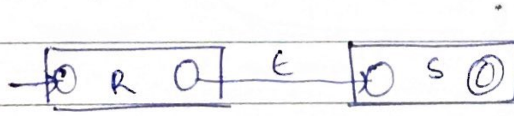
②  ϕ

③  a

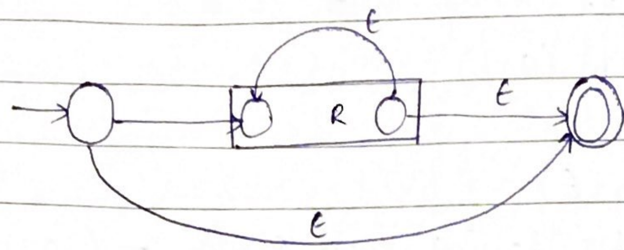
④ R+S



⑤ R·S



⑥ R^*

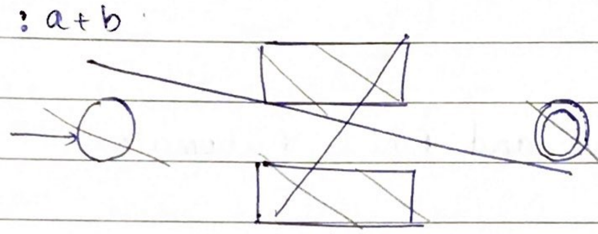


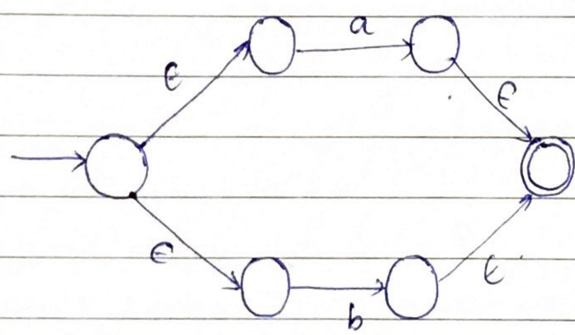
PROBLEMS :-

① $(a+b)^*$

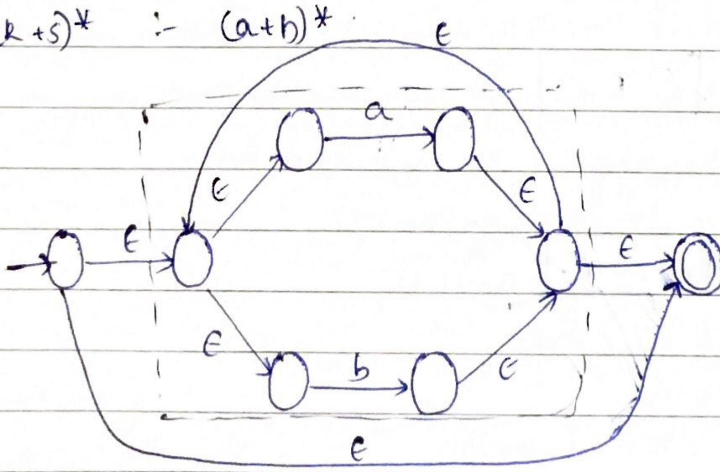
a) $R = a$, $S = b$

b) $R+S$: $a+b$





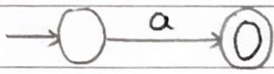
c) $(R+S)^*$:- $(a+b)^*$

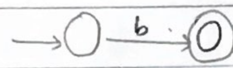


Page No.:
Date:

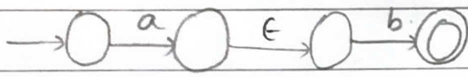
(9) ab (a + b)

R = a S = b

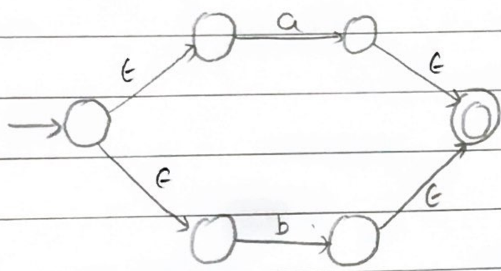




R.S = ab

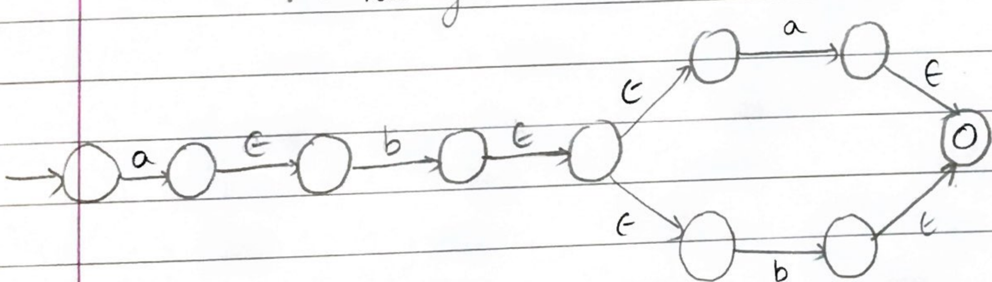


R + S = a + b



Now we consider ab to be R & (a+b) to be S.

∴ we get



Pg no. 60 (Ullman Text) (Theorem) (3rd edition)

If D is a DFA constructed from NFA N by the subset construction method, then $L(D) = L(N)$

SURYA Gold

Date_____ Page_____

Tutorial - 2

1. Costruct a DFA which accepts binary string divisible by 4. Show with the steps the input strings 10101 & 011000 are valid or not
2. Construct a DFA which accepts octal string divisible 3. 246 1357
3. Construct a DFA for the language $L = \{w \mid |w| \bmod 5 \neq 0\}$ $\Sigma = \{a, b\}$
4. Construct a DFA for the Language $L = \{w \mid |w| \bmod 4 \neq |w| \bmod 2\}$ $\Sigma = \{a, b\}$
5. Design a DFA to accept the strings of A's and B's such that the no of A's is divisible by 5 and the no of b's is divisible by 8
6. $L = \{w \mid n_a(w) \bmod 2 = 0 \;\&\; n_b(w) \bmod 2 = 0\}$
   $\Sigma = \{a, b\}$

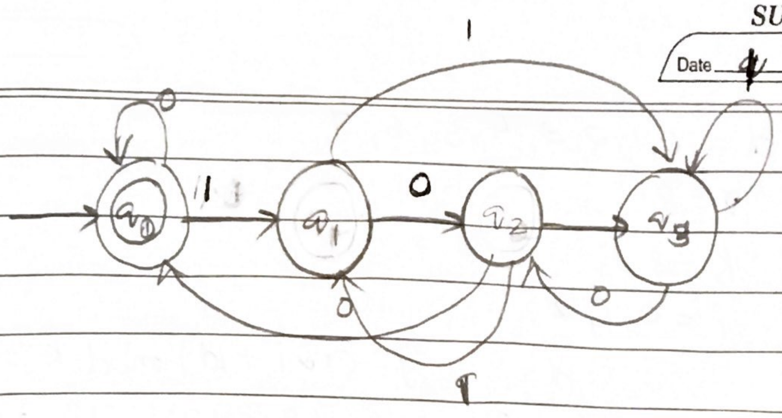
1. Here $i = 0, 1, 2, 3$

$k = 4$

$r = 2$

$d = 0, 1$

i	d	$j = (r * i + d) \bmod k$	q_j
0	0	$j = (2 * 0 + 0) \bmod 4 = 0$	q_0
0	1	$j = (2 * 0 + 1) \bmod 4 = 1$	q_1
1	0	$j = (2 * 1 + 0) \bmod 4 = 2$	q_2
	1	$j = (2 * 1 + 1) \bmod 4 = 3$	q_3
2	0	$j = (2 * 2 + 0) \bmod 4 = 0$	q_0
	1	$j = (2 * 2 + 1) \bmod 4 = 1$	q_1
3	0	$j = (2 * 3 + 0) \bmod 4 = 2$	q_2
	1	$j = (2 * 3 + 1) \bmod 4 = 3$	q_3



$Q = \{q_0, q_1, q_2, q_3\}$

$\Sigma = \{0, 1\}$

$q_0 = \{q_0\}$

$F = \{q_0\}$

$\delta(q_0, 0) = q_0$	$\delta(q_2, 0) = q_0$
$\delta(q_0, 1) = q_1$	$\delta(q_2, 1) = q_1$
$\delta(q_1, 0) = q_2$	$\delta(q_3, 0) = q_2$
$\delta(q_1, 1) = q_3$	$\delta(q_3, 1) = q_3$

$= \hat{\delta}(q_0, 10101)$

$= \hat{\delta}(q_1, 0101)$

$= \hat{\delta}(q_2, 101)$

$= \hat{\delta}(q_1, 01)$

$= \hat{\delta}(q_3, 1)$

$= \hat{\delta}(q_3, \epsilon)$

$\therefore$ q_3 is non final state it is invalid

$= \hat{\delta}(q_0, 011000)$

$= \hat{\delta}(q_0, 11000)$

$= \hat{\delta}(q_1, 1000)$

$= \hat{\delta}(q_3, 000)$

$= \hat{\delta}(q_2, 00)$

$= \hat{\delta}(q_0\ 0)$

$= \hat{\delta}(q_0, \epsilon)$

$\therefore$ It is valid, q_0 is final state

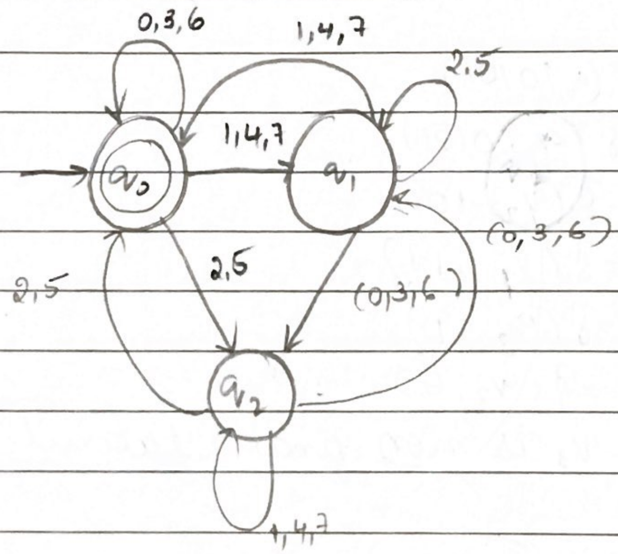
2. $d = 0,1,2,3,4,5,6,7$

$r = 8$

$k = 3$

$i = 0,1,2$

i	d	$j = (r \times i + d) \bmod k$	q_j
0	0 (0,3,6)	$j = (8\times 0+0) \bmod 3 = 0$	q_0
	1 (1,4,7)	$j = (8\times 0+1) \bmod 3 = 1$	q_1
	2 (2,5)	$j = 2$	q_2
1	(0,3,6)	$j = (8\times 1+0) \bmod 3 = 2$	q_2
	(1,4,7)	$j = (8\times 1+1) = 0$	q_0
	(2,5)	$j = (8\times 1+2) = 1$	q_1
2	(0,3,6)	$j = (8\times 2+0) = 1$	q_1
	(1,4,7)	$j = 2$	q_2
	(2,5)	$j = 0$	q_0



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

$\Sigma = \{0,1,2,3,4,5,6\}$

$F = \{q_0\}$

$q_0 = \{q_0\}$

$\delta \times q_0,$

(i) $\hat{\delta}(q_0, 246)$

$= \hat{\delta}(q_2, 46)$

$= \hat{\delta}(q_2, 6)$

$= \hat{\delta}(q_1, \epsilon)$

∴ not valid

(ii) $\hat{\delta}(q_0, 1357)$

$= \hat{\delta}(q_1, 357)$

$= \hat{\delta}(q_2, 57)$

$= \hat{\delta}(q_0, 7)$

$= \hat{\delta}(q_1, \epsilon)$

q_1 is non final state

∴ Invalid

3. Remainder: 0, 1, 2, 3, 4

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

$Q = \{q_0, q_1, q_2, q_3, q_4\}$

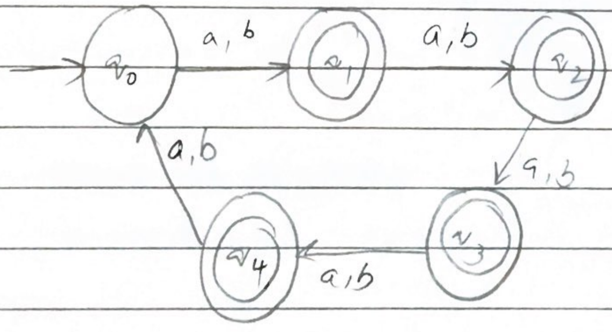
$\left.\begin{array}{l}\delta(q_0, a) \\ \delta(q_0, b)\end{array}\right\} = \cancel{q_0}\ \cancel{q_4}\ q_1$

$\left.\begin{array}{l}\delta(q_1, a) \\ \delta(q_1, b)\end{array}\right\} = q_2$

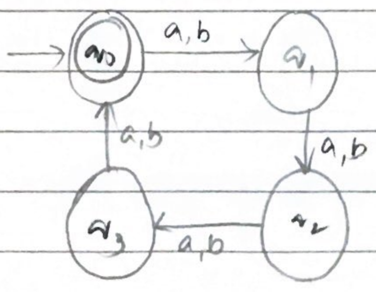
$\left.\begin{array}{l}\delta(q_2, a) \\ \delta(q_2, b)\end{array}\right\} = q_3$

$\left.\begin{array}{l}\delta(q_3, a) \\ \delta(q_3, b)\end{array}\right\} = q_4$

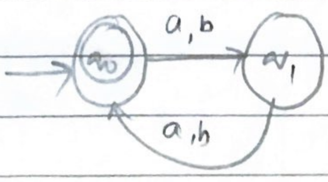
$\left.\begin{array}{l}\delta(q_4, a) \\ \delta(q_4, b)\end{array}\right\} = q_0$



4. D_1 mod 4



D_2 mod 2



$Q_1 = \{q_0, q_1, q_2, q_3\}$ $\quad$ $Q_2 = \{q_0, q_1\}$

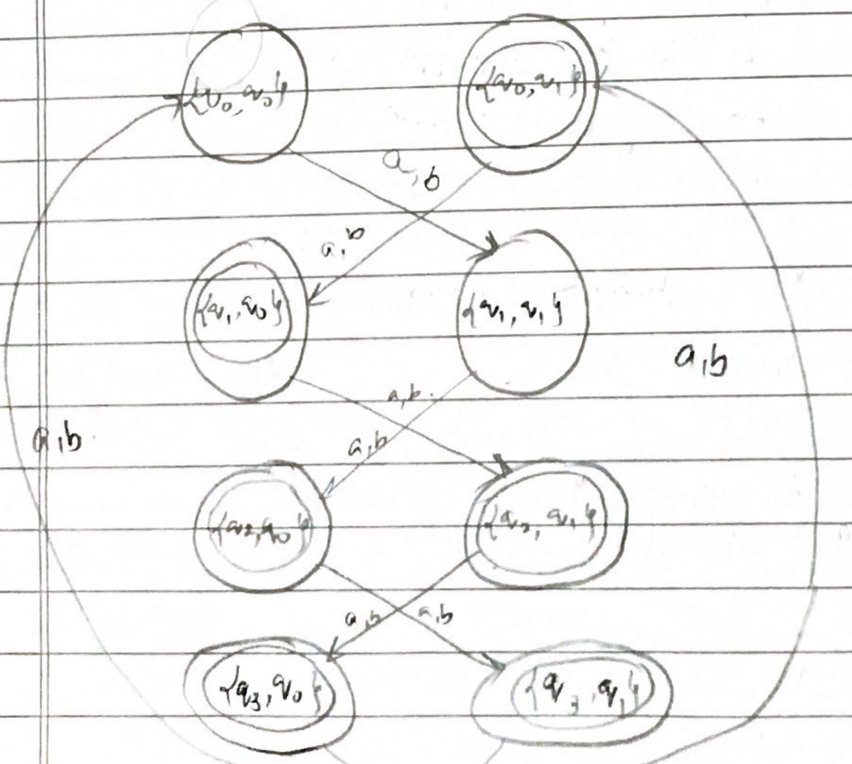
q_4 $\quad$ q_2

$Q = Q_1 \times Q_2$

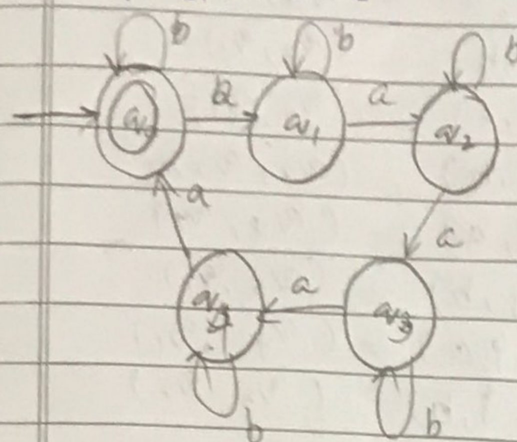
$= \{\{q_0, q_0\}, \{q_0, q_1\}, \{q_1, q_0\}, \{q_1, q_1\},$
$\{q_2, q_0\}, \{q_2, q_1\}, \{q_3, q_0\}, \{q_3, q_1\}\}$

$\{q_4, q_0\}, \{q_4, q_1\}, \{q_4, q_2\}$
$\{q_0, q_2\}, \{q_1, q_2\}, \{q_2, q_2\}$
$\{q_3, q_2\}$

$\delta(\{q_0, q_0\}, a) = \{q_1, q_1\}$
$\delta(\{q_0, q_0\}, b) = \{q_1, q_1\}$
$\delta(\{q_0, q_1\}, a) = \{q_1, q_0\}$
$\delta(\{q_0, q_1\}, b) = \{q_1, q_0\}$
$\delta(\{q_1, q_0\}, a) = \{q_2, q_1\}$
$\delta(\{q_1, q_0\}, b) = \{q_2, q_1\}$
$\delta(\{q_1, q_1\}, a) = \{q_2, q_0\}$
$\delta(\{q_1, q_1\}, b) = \{q_2, q_0\}$
$\delta(\{q_2, q_0\}, a) = \{q_3, q_1\}$ $\quad$ $\delta(\{q_2, q_0\}, b) = \{q_3, q_1\}$
$\delta(\{q_2, q_1\}, a) = \{q_3, q_0\}$
$\delta(\{q_2, q_1\}, b) = \{q_3, q_0\}$ $\quad$ δ
$\delta(\{q_3, q_0\}, a) = \{q_0, q_1\}$
$\delta(\{q_3, q_0\}, b) = \{q_0, q_1\}$
$\delta(\{q_3, q_1\}, a) = \{q_0, q_0\}$
$\delta(\{q_3, q_1\}, b) = \{q_0, q_0\}.$



5. D_1 a's mod 5



$\delta(\{q_4, q_0\}, a) = \{q_0, q_0\}$

$\delta(\{q_4, q_0\}, b) = \{q_4, q_1\}$

$\delta(\{q_4, q_1\}, a) = \{q_0, q_1\}$

$\delta(\{q_4, q_1\}, b) = \{q_4, q_2\}$

$\delta(\{q_4, q_2\}, a) = \{q_0, q_2\}$

$\delta(\{q_4, q_2\}, b) = \{q_4, q_0\}$

$\delta(\{q_0, q_2\}, a) = \{q_1, q_2\}$, $b = \{q_0, q_0\}$

$\delta(\{q_1, q_2\}, a) = \{q_2, q_2\}$, $b = \{q_1, q_0\}$

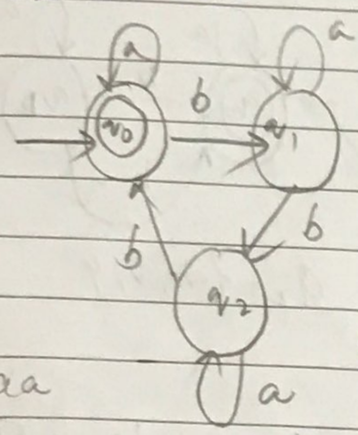
$\delta(\{q_1, q_2\}, a), b) = \{q_2, q_2\}$

$\delta(\{q_2, q_2\}, a) = \{q_3, q_2\}$, $b) = \{q_2, q_0\}$

$\delta(\{q_3, q_2\}, a) = \{q_4, q_2\}$

$\delta(\{q_3, q_2\}, b) = \{q_3, q_0\}$

D_2 b's mod 3



$\delta(\{q_0, q_0\}, a) = \{q_1, q_0\}$

$\delta(\{q_0, q_0\}, b) = \{q_0, q_1\}$

$\delta(\{q_0, q_1\}, a) = \{q_1, q_1\}$

$\delta(\{q_0, q_1\}, b) = \{q_0, q_2\}$

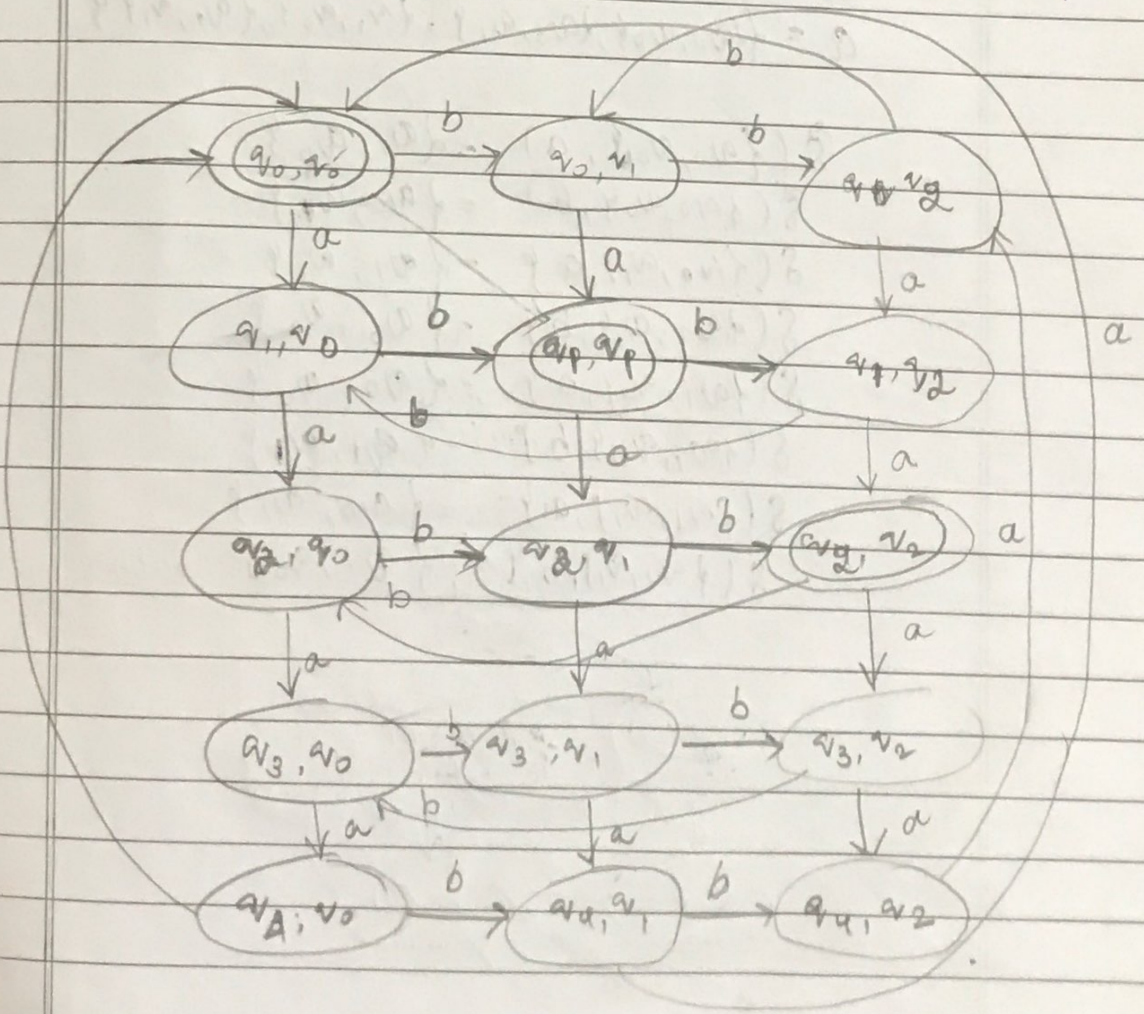
$\delta(\{q_1, q_0\}, a) = \{q_2, q_0\}$

$\delta(\{q_1, q_0\}, b) = \{q_1, q_1\}$

$\delta(\{q_1, q_1\}, a) = \{q_2, q_1\}$

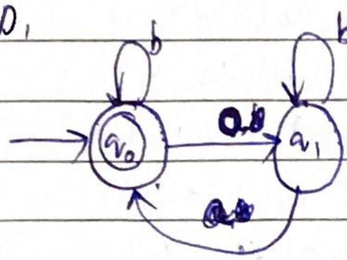
$\delta(\{q_1, q_1\}, b) = \{q_1, q_2\}$

aabbbaaa

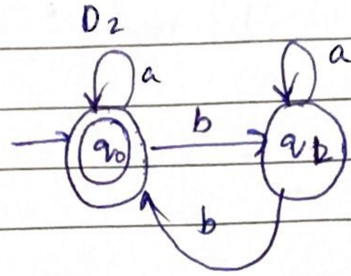


$\delta(\{q_2, q_0\}, a) = (q_3, q_0)$
$\delta(\{q_2, q_0\}, b) = (q_2, q_1)$
$\delta\{q_2, q_1\}, a) = (q_3, q_1)$
$\delta\{q_2, q_1\}, b) = (q_2, q_2)$
$\delta(\{q_3, q_0\}, a) = (q_4, q_0)$
$\delta(\{q_3, q_0\}, b) = (q_3, q_1)$
$\delta(\{q_3, q_1\}, a) = (q_4, q_1)$
$\delta(\{q_3, q_1\}, b) = (q_3, q_2)$

6. D_1



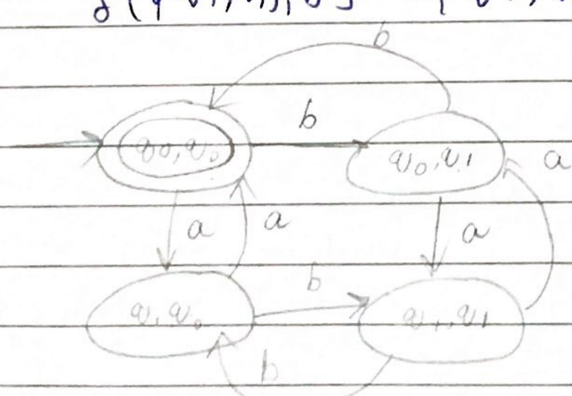
D_2



$Q_1 = \{q_0, q_1\}$ $\quad$ $Q_2 = \{q_0, q_1\}$

$Q = \{Q_1 \times Q_2$
$Q = \{\{q_0, q_0\}, \{q_0, q_1\}, \{q_1, q_0\}, \{q_1, q_1\}\}$

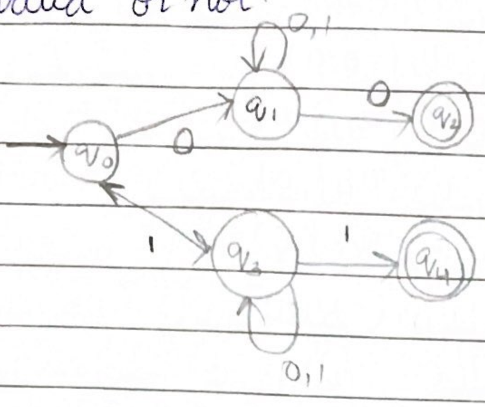
$\delta(\{q_0, q_0\}, a) = \{q_1, q_0\}$
$\delta(\{q_0, q_0\}, b) = \{q_0, q_1\}$
$\delta(\{q_0, q_1\}, a\} = \{q_1, q_1\}$
$\delta(\{q_0, q_1\}, b\} = \{q_0, q_0\}$
$\delta(\{q_1, q_0\}, a\} = \{q_0, q_0\}$
$\delta(\{q_1, q_0\}, b\} = \{q_1, q_1\}$
$\delta(\{q_1, q_1\}, a\} = \{q_0, q_1\}$
$\delta(\{q_1, q_1\}, b\} = \{q_1, q_0\}$ aabba



Tutorial - 3

1. construct NFA for strings of 0's & 1's with same first & last symbol write $(Q, \Sigma, q, \delta, F)$ show with the steps if strings 1010 & 11001 are valid or not.

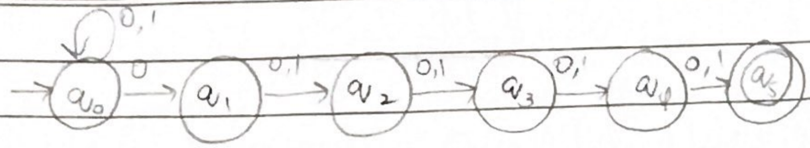
→



(i) $\hat{\delta}(q_0, 1010)$
$= \hat{\delta}(q_3, 010)$
$= \hat{\delta}(\{q_3\}, 10)$
$= \hat{\delta}(\{q_3, q_4\}, 0)$
$= \hat{\delta}(q_3, \epsilon)$
$\therefore$ q_3 is non final state
It is invalid

(ii) $\hat{\delta}(q_0, 11001)$
$= \hat{\delta}(q_3, 1001)$
$= \hat{\delta}(\{q_3, q_4\}, 001)$
$= \hat{\delta}(\{q_3, q_4\}, 01)$
$= \hat{\delta}(\{q_3, q_4\}, 1)$
$= \hat{\delta}(\{q_3, q_4\}, \epsilon)$
q_4 is final state
valid

2. Design NFA for strings 0's & 1's with a 0 as 5^{th} last symbol. Write $(Q, \Sigma, \delta, q, F)$ show if 01010 of 011001 are valid or not.



→ (i) $\omega = 01010$

$\hat{\delta}(q_0, 01010)$

$= \hat{\delta}(\{q_0, q_1\}, 1010)$

$= \hat{\delta}(\{q_0, q_2\}, 010)$

$= \hat{\delta}(\{q_0, q_1, q_3\}, 10)$

$= \hat{\delta}(\{q_0, q_2, q_4\}, 0)$

$= \hat{\delta}(\{q_0, q_3, q_5\}, \epsilon)$

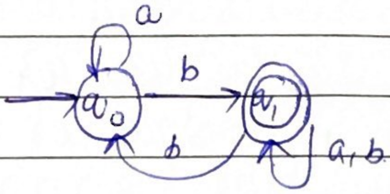
q_5 is final state

∴ Valid

3. construct DFA for the given NFA using subset constructor method.

(i)

δ	a	b
$\rightarrow q_0$	$\{q_0\}$	$\{q_1\}$
$* q_1$	$\{q_1\}$	$\{q_0, q_1\}$



→ subsets of $Q = \{\phi, \{q_0\}, \{q_1\}, \{q_0, q_1\}\}$

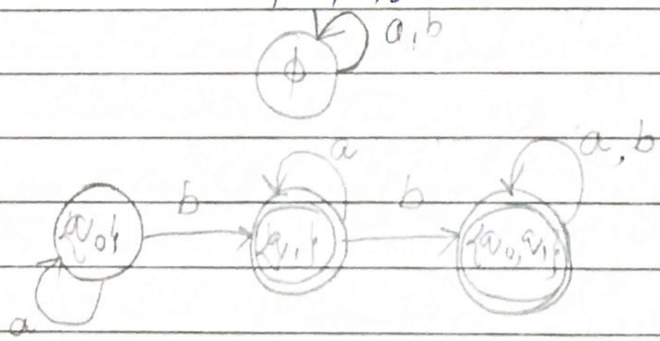
$\delta(\phi, a) = \phi$ $\quad\quad$ $\delta(\phi, b) = \phi$

$\delta(\{q_0\}, a) = \{q_0\}$ $\quad\quad$ $\delta(\{q_0\}, b)\ \{q_1\}$

$\delta(\{q_1\}, a) = \{q_1\}$ $\quad\quad$ $\delta(\{q_1\}, b) = \{q_1, q_0\}$

$\delta(\{q_0, q_1\}, a) = \{q_0\} \cup \{q_1\}$ $\quad\quad$ $\delta(\{q_0, q_1\}, b) = \{q_1 \cup q_0\}$

$= \{q_0, q_1\}$



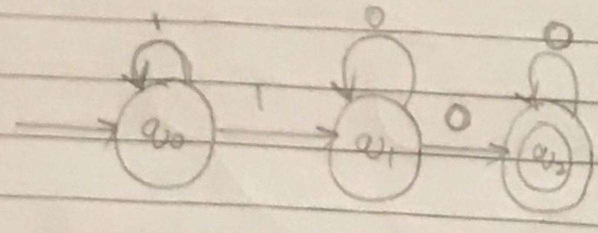
SURYA Gold

Date ______ Page ______

(ii)

δ	0	1
$\rightarrow q_0$	ϕ	$\{q_0, q_1\}$
q_1	$\{q_1, q_2\}$	ϕ
$* q_2$	$\{q_2\}$	ϕ

$\rightarrow$



Subsets of $Q = \{\phi, \{q_0\}, \{q_1\}, \{q_2\}, \{q_0, q_1\}, \{q_1, q_2\}, \{q_0, q_2\}, \{q_0, q_1, q_2\}\}$

$\delta(\phi, 0) = \phi$ $\qquad \delta(\phi, 1) = \phi$

$\delta(\{q_0\}, 0) = \phi$ $\qquad \delta(\{q_0\}, 1) = \{q_0, q_1\}$

$\delta(\{q_1\}, 0) = \{q_1, q_2\}$ $\qquad \delta(\{q_1\}, 1) = \phi$

$\delta(\{q_2\}, 0) = \{q_2\}$ $\qquad \delta(\{q_2\}, 1) = \phi$

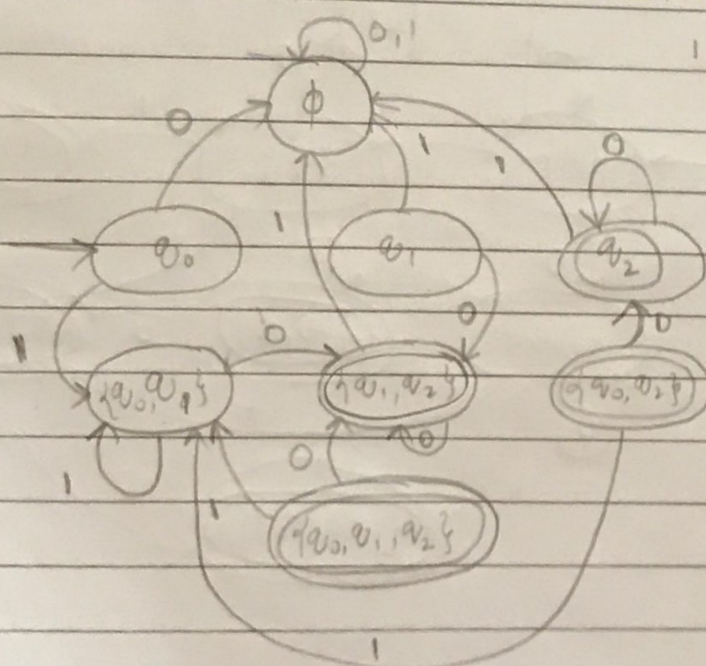
$\delta(\{q_0, q_1\}, 0) = \{q_1, q_2\}$ $\qquad \delta(\{q_0, q_1\}, 1) = \{q_0, q_1\}$

$\delta(\{q_1, q_2\}, 0) = \{q_1, q_2\}$ $\qquad \delta(\{q_1, q_2\}, 1) = \phi$

$\delta(\{q_0, q_2\}, 0) = \{q_2\}$ $\qquad \delta(\{q_0, q_2\}, 1) = \{q_0, q_1\}$

$\delta(\{q_0, q_1, q_2\}, 0) = \{q_1, q_2\}$ $\qquad \delta(\{q_0, q_1, q_2\}, 1) = \{q_0, q_1\}$

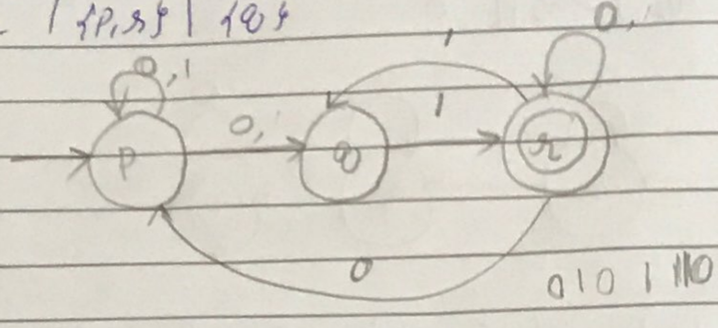
11000



(iii)

δ	0	1
$\rightarrow p$	$\{p, q\}$	$\{p\}$
q	ϕ	$\{r\}$
$* \; r$	$\{p, r\}$	$\{q\}$

$\rightarrow$



subset of $Q = \{\phi, \{p\}, \{q\}, \{r\}, \{p, q\}, \{q, r\},$
$\{p, r\}, \{p, q, r\}\}$

$\delta(\phi, 0) = \phi$ $\quad$ $\delta(\phi, 1) = \phi$

$\delta(\{p\}, 0) = \{p, q\}$ $\quad$ $\delta(\{p\}, 1) = \{p\}$

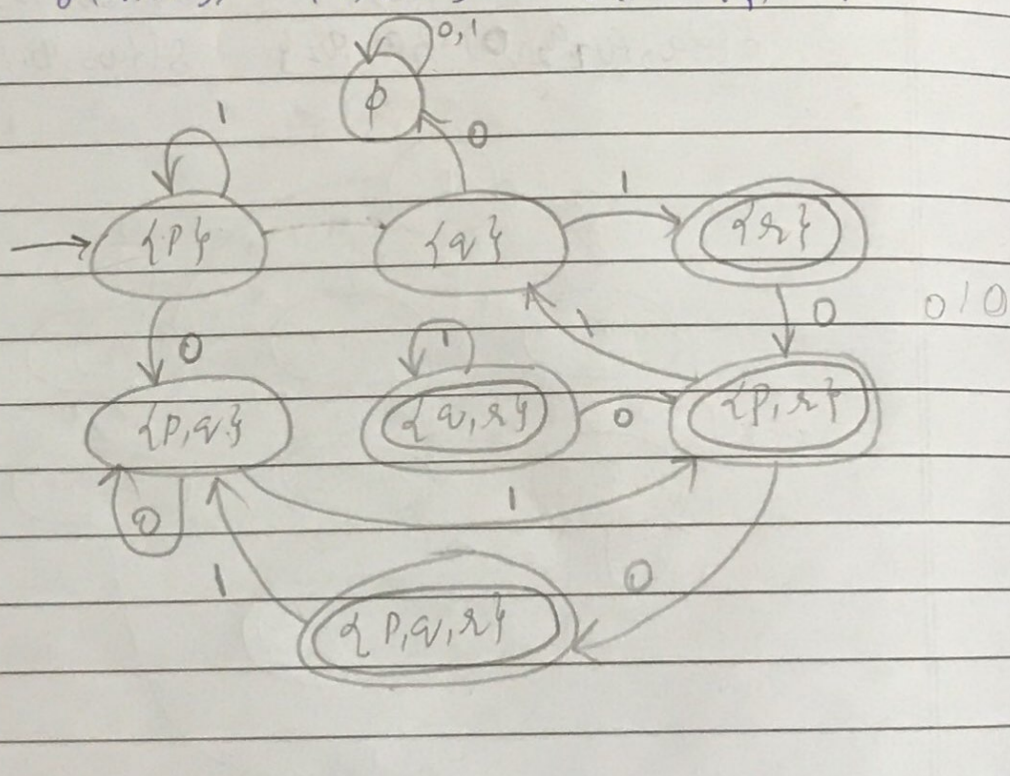
$\delta(\{q\}, 0) = \{\phi\}$ $\quad$ $\delta(\{q\}, 1) = \{r\}$

$\delta(\{r\}, 0) = \{p, r\}$ $\quad$ $\delta(\{r\}, 1) = \{q\}$

$\delta(\{p, q\}, 0) = \{p, q\}$ $\quad$ $\delta(\{p, q\}, 1) = \{p, r\}$

$\delta(\{q, r\}, 0) = \{p, r\}$ $\quad$ $\delta(\{q, r\}, \phi) = \{r, q\}$

$\delta(\{p, r\}, 0) = \{p, q, r\}$ $\quad$ $\delta(\{p, r\}, \phi) = \{p, q\}$



FA & RE

I.) RE to FA : E-NFA construcⁿ

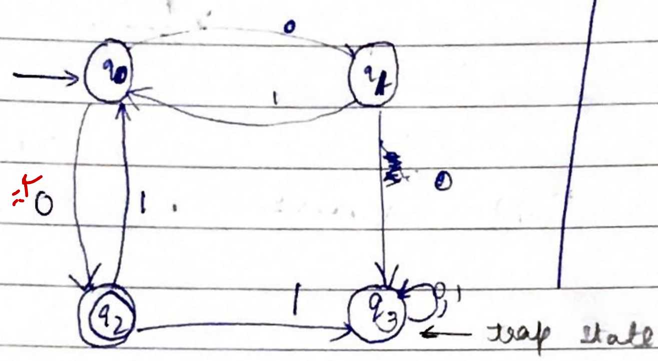
II.) FA to RE : ① By eliminating states
② R_{ij}^{k} method (Kleene Theorem)

Problems:

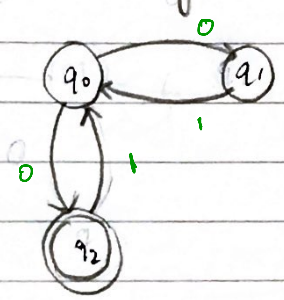
Q) Obtain the R.E for the following FA

① (a)

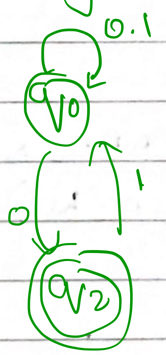
Solⁿ



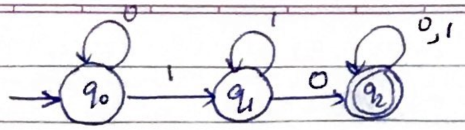
After eliminating trap state - q_3.



eliminating q_1

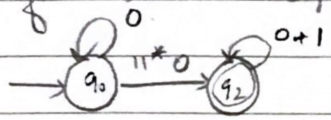


Q)



Soln

(i) After eliminating q_1



$R = 0$

$S = 11^*0$

$T = \phi$

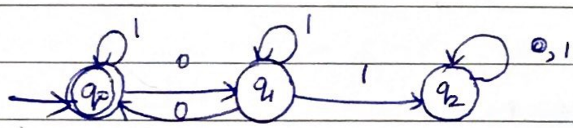
$U = 0+1$

$$RE = (0 + 11^*0(0+1)^*\phi)^* 11^*0(0+1)^*$$
$$= 0^* 11^* 0 (0+1)^*$$

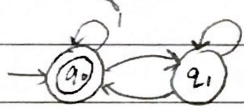
Imp

$$\boxed{\begin{array}{l} \phi^* = \epsilon \\ (w \cdot \phi) = \phi \\ (w \cdot \epsilon) = w \end{array}}$$

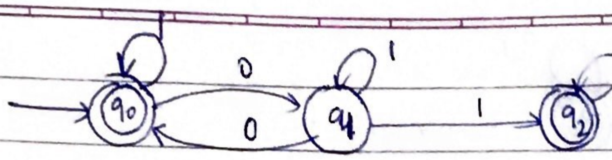
3)



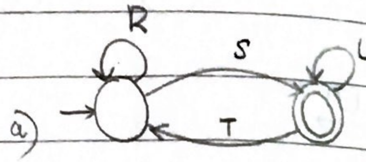
Soln (i) After eliminating q_2



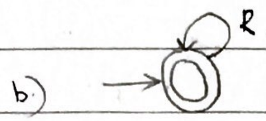
4)



Note



$R \cdot E = (R + SU^* T)^* SU^*$



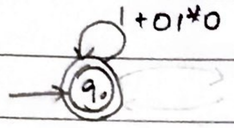
$R \cdot E = R^*$

Sol^n →

$q_0 (FS) \longrightarrow R_1$

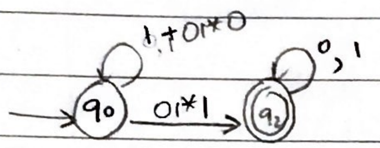
$q_2 (FS) \longrightarrow R_2$

$R_1 \rightarrow$



$R \cdot E = (1 + 01^*0)^*$

$R_2 \rightarrow$



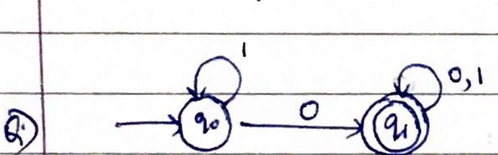
$R \cdot E = ((1+$

II **Kleene's theorem** :-
[R_{ij}^k method].

intermediate state no's should not be $< k$. (It can only have $\leq k$ intermediate states)

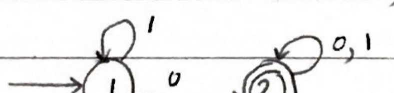
$$R_{ij}^k = R_{ij}^{k-1} + R_{ik}^{k-1} (R_{kk}^{k-1})^* R_{kj}^{k-1}$$

Q.



Sol^n →

① Rename the states as 1, 2, 3, 4 - - -



i → source

j → destination

k → intermediate states

R → Regular expr^n

Basis :-

② $R_{11}^{(0)} = 1 + \epsilon$

$R_{12}^{(0)} = 0$

$R_{21}^{(0)} = \phi$

$R_{22}^{(0)} = 1 + 0 + \epsilon$

Page No.:
Date:

③ $R_{11}^{(1)} = R_{11}^{(0)} + R_{11}^{(0)}(R_{11}^{(0)})^* R_{11}^{(0)}$

$= 1+\epsilon + (1+\epsilon)(1+\epsilon)^* (1+\epsilon)$

$R_{11}^{(1)} = 1^*$

$R_{12}^{(1)} = R_{12}^{(0)} + R_{11}^{(0)} (R_{11}^{(0)})^* R_{12}^{(0)}$

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

$= 0 + 1^*0 = 1^*0$

$R_{21}^{(1)} = R_{21}^{(0)} + R_{21}^{(0)} (R_{11}^{(0)})^* R_{11}^{(0)}$

$= \phi + \phi (1+\epsilon) (1+\epsilon)$

$= \phi$

$R_{22}^{(1)} = R_{22}^{(0)} + R_{21}^{(0)} (R_{11}^{(0)})^* R_{12}^{(0)}$

$= (1+0+\epsilon) + \phi (1+\epsilon)^* 0$

$= (0+1+\epsilon)$

④ $R_{11}(2) = R_{11}(1) + R_{12}^{(1)}(R_{22})^* R$

$= 1^* + 1^*0 (1+1+\epsilon)^{*12} \phi$

$= 1^* + 1^*0 (0+1+\epsilon)^* \phi$

$R_{11}(2) = 1^*$

no. of state

$R_{12}^{(2)} = R_{12}^{(1)} + R_{12}^{(1)} (R_{22}^{(1)})(R^{(1)}_{22})^* (R_{22}^{(1)})]$

source destination

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

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

$= 1^*1^*0 + 1^*0 (0+1)^*$

$= 1^* 0(0+1)^*$

no of states

R source destination

stop vielts

Page No.:

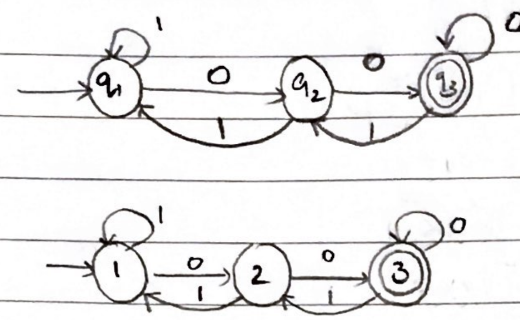
Date:

for text answer

2.

	0	1
$\rightarrow q_1$	q_2	q_1
q_2	q_3	q_1
$* q_3$	q_3	q_2

solⁿ →



$R_{13}^{(3)}$

Basis :-

$R_{11}^{(0)} = 1 + \epsilon$ $\qquad$ $R_{13}^{(0)} = \phi$

$R_{12}^{(0)} = 0$

$R_{21}^{(0)} = 1$ $\qquad$ $R_{23}^{(0)} = 0$

$R_{22}^{(0)} = \epsilon$

$R_{31}^{(0)} =$ $\qquad$ $R_{33}^{(0)} =$

$R_{32}^{(0)} =$

Minimised DFA!

Steps:

1) Identify the distinguishable & indistinguishable states [using table-filling algorithm].
2) Find the states of minimised DFA.
3.) Complete the transition table.
4) Identify initial & final states.

Ex 1)

δ	0	1
$\rightarrow$ A	B	F
B	G	C
* C	A	C
D	C	G
E	H	F
F	C	G
G	G	E
H	G	C

① table filling algorithm:

–	X	X
X	X	X
X	–	X

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

②

	0	1	
(A, B)	(B, G)	(F, C)	(F, C) is marked, so mard (A, B).
(A, D)	(B, C)	(F, G)	(B, C) is marked, so mark (A, D)
(A, E)	(B, H)	(F, F)	don't mark
(A, F)	(B, C)	(F, G)	(B, C) is marked, so mark (A, F)
(A, G)	(B, G)	(F, E)	don't mark.
(A, H)	(B, G)	(F, C)	(F, C) is marked, so mark (A, H)

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

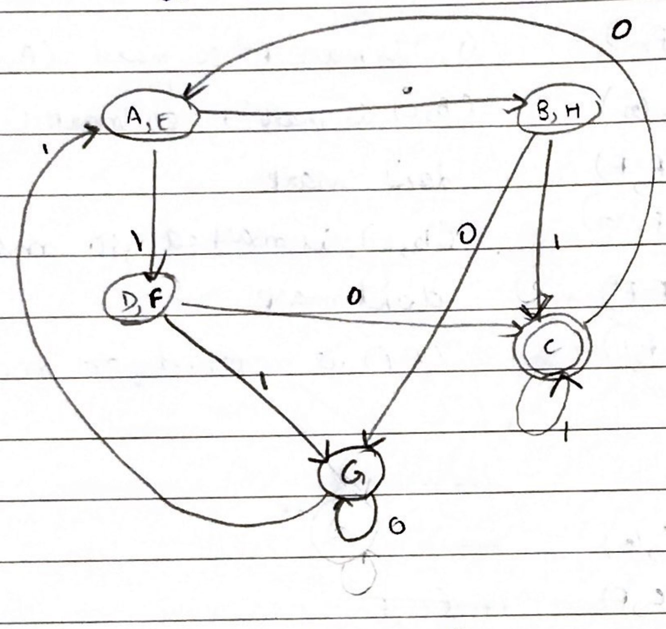
	0	1
(D, E)	(C, H)	(G, F)
(D, F)	(C, C)	(G, G)
(D, G)	(C, G)	(G, E)
(D, H)	(C, G)	(G, C)

	0	1.
(E, F)	(H, C)	(F, G)
(E, G)	(H, G)	(F, E)
(E, H)	(H, G)	(F, C).

	0	1
③ (A, E)	(B, H)	(F, E)
(A, G)	(B, G)	(F, E)
(B, H)	(G, G)	(C, C)
(D, F)	(C, C)	(G, G)

TT for minized DFA

	0	1
→ (A, E)	(B, H)	(D, F) ⟶ (F, E)
(B, H)	~~(G, G)~~ G	~~(C, C)~~ C
(D, F)	~~(C, C)~~ C	~~(G, G)~~ G.
* C	(A, E)	C
G	G	(A, E).



Minimize the following DFA.

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

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

Page No.:
Date:

(2)

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

~~(C, D)~~

(C, F)	(D, G)	(H, B)
~~(C, G)~~	(	~~(H, E)~~
(C, I)	(D, A)	(H, E)
(D, E)	(E, F)	(H, I)
(D, G)	(E, H)	(H, B)
(D, H)	(E, I)	(H, C)
(E, G)	(F, H)	(I, B)
(E, H)	(F, I)	(I, C)
(F, J)	(G, A)	(B, E)
(G, H)	(H, I)	(B, C)

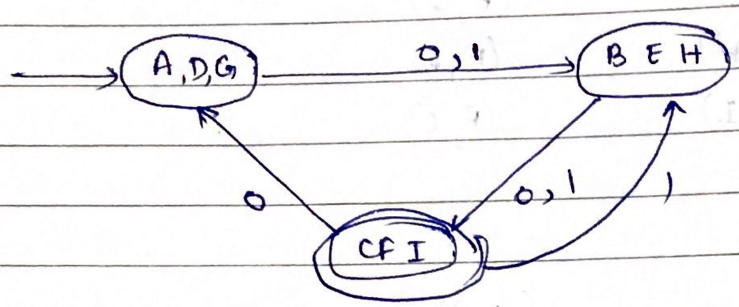
(3)

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

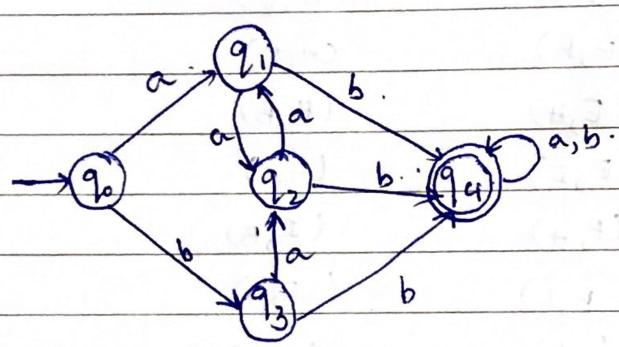
Page No.:
Date:

Transition table of minimized DFA

	0	1
→ (A, D, G)	(B, E, H)	(E, H, B)
(B, E, H)	(C, F, I)	(F, I, C)
* (C, F, I)	(D, G, A)	(H, B, E)



Q)



	a	b
→ q_0	q_1	q_3
q_1	q_2	q_4
q_2	q_1	q_4
q_3	q_2	q_4
* q_4	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

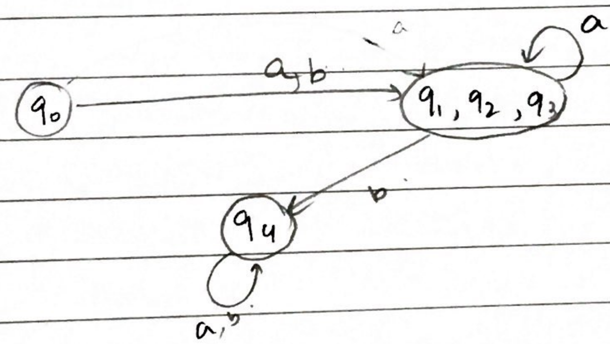
	a	b
(q_0, q_1)	(q_1, q_2)	$(q_3\ q_4)$
(q_0, q_2)	(q_1, q_1)	(q_3, q_4)
(q_0, q_3)	(q_1, q_2)	(q_3, q_4)

~~q_2~~	a	b
(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)		

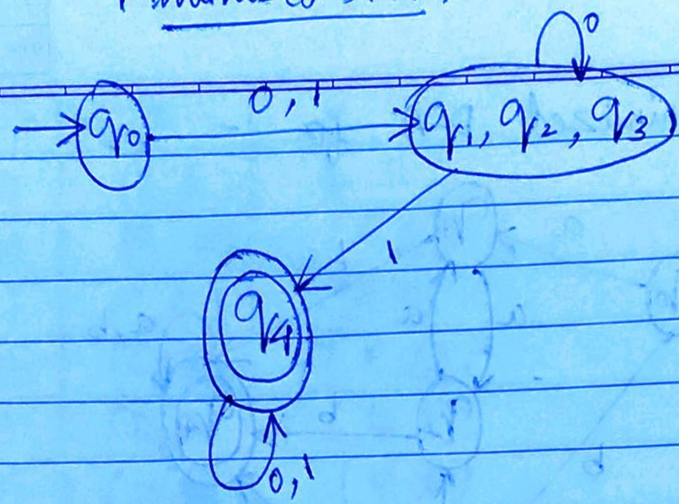
	a	b
(q_1, q_2)	(q_2, q_1)	(q_4, q_4)
(q_1, q_3)	(q_2, q_2)	(q_4, q_4)
(q_2, q_3)	(q_2, q_2)	(q_4, q_4)
(q_4, q_4)	(q_4, q_4)	(q_4, q_4)

(3)

TT	a	b
$\rightarrow q_0$	(q_1, q_2, q_3)	(q_3, q_1, q_2)
$\rightarrow (q_1, q_2, q_3)$	(q_2, q_3, q_1, q_2)	(q_4, q_4, q_4)
$*(q_4)$	(q_4)	(q_4)



Minimized DFA :-



Valid String :- 01

Invalid :- 00

26/3/2018

Unit III (Context free Language)

L → FA, RE

Pumping Lemma Theorem :- (Regular Language)!

Let L be a Regular Language, then, there exists a constant n, such that for every string w, $|w| \geq n$, we can break w into 3 strings x, y, z such that

where n → ~~Length of String w~~ No of states

$|w|$ — String Length

i) $y \neq \in$

ii) $|xy| \leq n$

iii) For all $k > 0$, xy^kz is also in L

Only to say Given Language is not a regular Language.

$a^n b$ → FA ✓, RE ✓

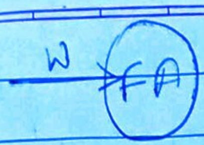
~~$a^n b^n$~~ → FA ✗, RE ✗

Not a Regular Language

Not same as a^*b^*

used for dividing the string

$w = xyz$

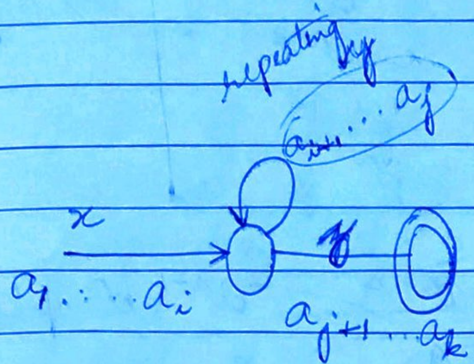


Pigeon hole principle (Concept)

$\hookrightarrow n > m$

m — pigeon holes

n — pigeons.



$W = xy^k z$ ✓

$k=0$; xz ✓

$k=1$; xyz ✓

$k=2$; xy^2z ✓

Note :-

* Theorem proofs read from T.B : [6 — 8 marks]

Problems :-

1) S.T $a^n b^n$ is non regular language.

A) Applying pumping Lemma Theorem,

$L = a^n b^n$; $n=2$ then a^2b^2

$W = aabb$

$x = a$	$x = \epsilon$	$x = \epsilon$	Conditions :-
$y = a$	$y = aa$	$y = a$	i) $y \neq \epsilon$
$z = bb$	$z = bb$	$z = abb$	ii) $\lvert xy \rvert \leq n$
			iii)

xy^kz

$k=0$; $abb \notin L$

↳ Not a Regular Language

$k=0$, $xz \Rightarrow bb \notin L$

Not a Regular Language

$k=0$

$xz \Rightarrow abb \notin L$

Not a RL

Note :- If $K=0$, $ab \in L$

$K=1$, $aabb \in L$

$K=2$, $abb \notin L$

Not a RL

Answer 1

2) S.T $a^i b^j \mid i > j$ is a Non Regular Language.

$w = a^n b^n$ $\quad a^{n-2} b^n \notin L \quad a^{n-3} b^n \notin L$

| i) $y \neq \epsilon$
| ii) $|xy| \leq n$

$x = a^{n-1} \quad x = a^{n-2} \quad x = a^{n-3}$

$y = a \quad y = a^2 \quad y = a^3$

$z = b^n \quad z = b^n \quad z = b^n$

$k = 0$, xz, $a^{n-1} b^n \notin L$

$a^n b^n$ is not RL

$x = a^{n-2}$ — $a^{n-1} b^n \notin L$

$y = a$

$z = ab^n$

Ans) $w = a^{n+1} b^n \mid i > j$

$x = a$

$y = a^{n-1}$

$z = ab^n$

for $k = 0$, xz, $aab^n \notin L$

2) S.T the language $L = \{ w \mid n_a(w) < n_b(w) \}$ is a Non regular language.

$w = a^{n-1} b^n$

$x = a^{n-2}$

$y = a$

$z = b^n$

| i) $y \neq \epsilon$
| ii) $|xy| \leq n$

$w = xy^k z$

if $k = 0$, $xz \rightarrow a^{n-2} b^n \in L$

$k = 1$, $xyz \rightarrow a^{n-1} b^n \in L$

$k = 2$, $xy^2 z \rightarrow a^{n-2} \cdot a^2 \cdot b^n$

$\Rightarrow a^n b^n \notin L$

So given language is a Non Regular Language.

Page No.:
Date:

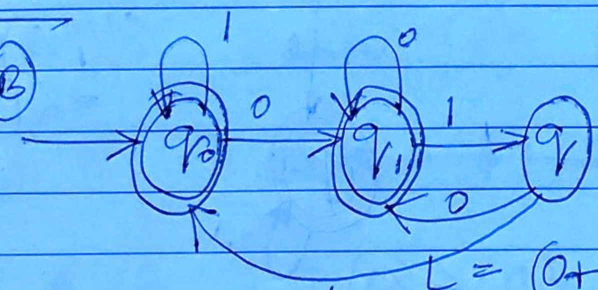
27/03/2018 * With diagram, Practice the Theorem for TB

Closure properties of RL's

1. Union of 2RL's is regular
2. Complement of a RL is regular
3. Intersection of 2 RLs is regular
4. Difference — " —
5. Reversal of a RL is regular
6. Closure of a 2RL's is regular
7. Concatenation of 2RL's is regular
8. Homomorphism of RL is regular
9. Inverse homomorphism of RL is regular.

Problem :-

Theorem 2 → (B)



Not ending with 01

F → NF
NF → F

$L = (0+1)^* 01$

$\Sigma^* = (0+1)^*$

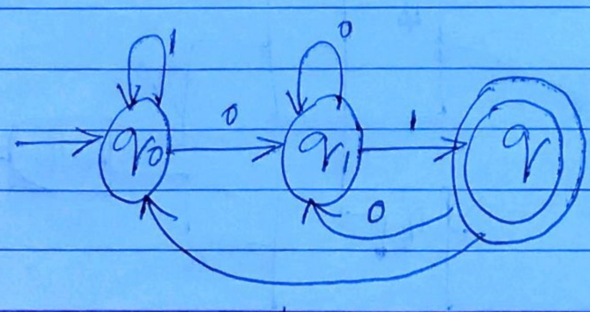
$\Sigma = \{0, 1\}$

$\bar{L} = \Sigma^* - L$

(set of all strings that accepts 0 & 1

$A = \{Q, \Sigma, \delta, q_0, F\}$
$B = \{Q, \Sigma, \delta, q_0, (Q-F)\}$

(A)



* RL ⇒ RE ⇒ ε-NFA ⇒ DFA ⇒ complement DFA

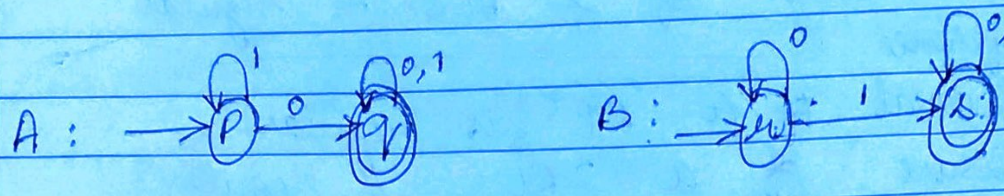
⇓

Eliminating states or R_{ij}^{k}

RE ⇐

Proof :-

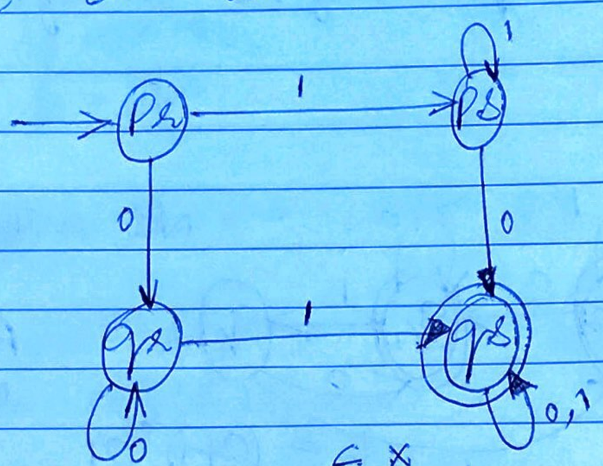
* ③ Intersection of 2 RL is a regular language.



When we take intersection, it should be strings of atleast one 0 & one 1

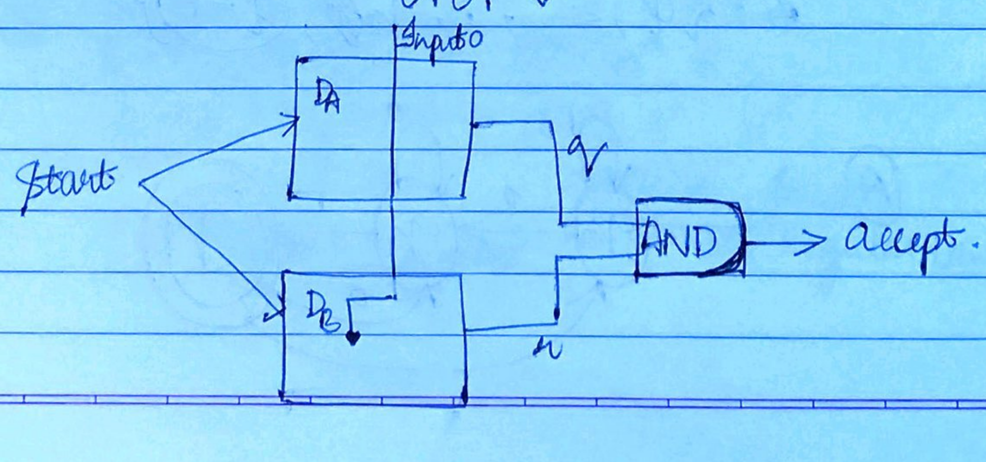
$Q = Q_A \times Q_B$

Commonly asked



$\delta(\{p,s\},0\}$
$= \{(p,0)(s,0)\}$
$\Rightarrow \{q,s\}$

ε ×
10 ✓
11 ×
01 ✓
0 ×
1 ×
0101 ✓



Page No.:
Date:

$A = \{Q_A, \Sigma, \delta_A, q_{0A}, F_A\}$ $B = \{Q_B, \Sigma, \delta_B, q_{0B}, F_B\}$

$C = \{Q_A \times Q_B, \Sigma, \delta_C, (q_{0A}, q_{0B}), (F_A, F_B)\}$

DFA definition of intersected DFA $\longrightarrow$

④ Difference of 2 RL is a regular expression.

By demorgan Theorem :-

$R - S = R \cap \overline{S}$

We know R is regular
$\overline{R}$ is regular

- $\overline{S}$ is also RL
- Intersection of 2RL is a RL

$\Rightarrow$ Hence proved.

⑤ Reversal of a RL is a RL.

RE definition :-

reverse

i) $R = \epsilon$; $L(R) = \{\epsilon\}$; $L(R^R) = \{\epsilon\}$

ii) $R = \phi$; $L(R) = \{\phi\}$; $R^R = \phi$, $L(R^R) = \{\phi\}$

iii) $R = a$; $L(R) = \{a\}$; $R^R = a$, $L\{R^R\} = \{a\}$

iv) $R+S$, $R = a+b = \{a, b\}$
$R^R = (a+b)^R = \{a, b)$

$L\{R^R\} = \{a, b\}$

v) $X = R.S$; $X^R = \underline{S^R . R^R}$

Eg: $R = ab$ $RS = abcd$
$S = cd$ $(RS)^R = S^R R^R = dcba$

vi) R^* ~~= R~~

$(R^R)^* = $

$R = ab$ ~~$(RS)^* = (abcd)^*$~~

~~$S = cd$~~ ~~$(R^R)^* =$~~

$R^* = (ab)^*$

$(R^R)^* = (ba)^*$

Theorem

1) Union of 2 RL's is regular.

$L_1 \Rightarrow R_1$ $\quad L(R_1) = L_1$

$L_2 \Rightarrow R_2$ $\quad L(R_2) = L_2$

RE defn union of 2 RE is RE

$R_1 + R_2 \Rightarrow R$

$L(R) = L_3$

$L(R_1 + R_2) \Rightarrow L(R)$

28/03/2018

Page No.:
Date:

Unit-3

Context free Grammer (CFG)

$G = \{V, T, P, S)$

V: Set of Variables (uppercase)
T: Set of Terminals
P: Set of productions /rules
S: Starting variables

Each production must be of the form :-

$V \rightarrow (V \cup T)^*$ → combination of variables & terminals

Ex:-

$S \rightarrow A$ — ①
$A \rightarrow aA / b$ — ②

R.E = a^*b

$V = \{S, A\}$
$T = \{a, b\}$

$S \rightarrow A$
$A \rightarrow aA$
$A \rightarrow b$

$S \Rightarrow A$
$\Rightarrow aA$
$\Rightarrow aaA$
(Terminal) aab

language ⇒ $L = \{S_1, S_2, S_3 \ldots S_n\}$
i.e set of all strings

1) $S \Rightarrow A$
$\Rightarrow b$ (Terminal)

2) $S \Rightarrow A$
$\Rightarrow aA$
$\Rightarrow ab$

So $L = \{b, ab, aab, aaaab \ldots a^nb\}$

Called → once
$S \rightarrow A$
$A \rightarrow aA \mid b$ once
n times

For any given FA, we can write the CFG

3) $(a+b)^*$ → FA — DFA, NFA, ∈-NFA

CFG

Non regular

$a^n b^n$ → CFG

Push down automata

- R.E or Non R.E we can write CFG
- only for R.E we can write finite automata

RL's { RE → CFG, FA → CFG }

- for non R.E we have special automata called push down automata

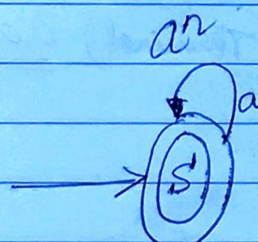
I CFG for RL's :-

↳ regular language

a) FA to CFG :-

Obtain CFG to generate string consisting of any no of a's ie 0 or more a

1) a^n ; $n \geq 0$



Transition Function :-

i) $\delta(S, a) = S$

$S \rightarrow aS$ ——— ① Rule no 1

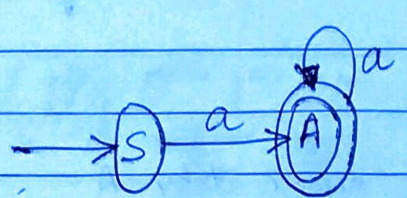
$S \rightarrow \epsilon$

* for having a terminal ending state.

$S \rightarrow aS \mid \epsilon$

$L = \{\epsilon, a, aa, \dots a^n\}$

2) atleast one 'a'



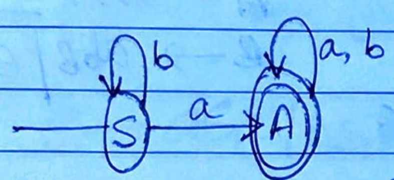
a) $\delta(S, a) = A$
$\delta(A, a) = A$

$\equiv$ $S \rightarrow aA$
$\equiv$ $A \rightarrow aA / \epsilon$

2 Rule as there are 2 states.

$L = \{a, aa, aaa, \dots \dots a^n\} ; n \geq 1$

3) Any number of a's and b's with atleast one 'a'



a) $\delta(S, a) = A \equiv S \rightarrow aA$
$\delta(S, b) = S \equiv S \rightarrow bS$
$\delta(A, a) = A \equiv A \rightarrow aA$
$\delta(A, b) = A \equiv A \rightarrow bA$

as A is a (Final State) $A \rightarrow \epsilon$

$S \rightarrow aA / bS$
$A \rightarrow aA / bA / \epsilon$
$L = \{a, \dots \quad\}$

Page No.:
Date:

II ~~b)~~ Regular expression (RE) to CFG :-

① a) Strings ~~of~~ As and Bs ending with ab

$\underbrace{(a+b)^*}_{A}ab$

Wherever we come across *, replace it by A

$S \to A\cancel{*}ab$
$A \to aA / bA / \epsilon$ because of closure

~~②~~ b) $(a+b)^* (ab)^*$

$\downarrow$ A $\quad \downarrow$ B

~~$S \to Aab$~~
~~$A \to aA / bA / \epsilon$~~

$S \to AB$
$A \to$ ~~///~~
$B \to abB / \epsilon$

③ Strings ~~of~~ As and Bs having substring 'ab

$\underbrace{(a+b)^*}_{A}ab\underbrace{(a+b)^*}_{A}$

$S \to AabA$
$A \to aA / bA / \epsilon$

Regular language { FA, RE } → CFG

III CFG for non-regular languages.

① ⊙ $L = \{ a^n b^n, n \geq 0 \}$

for a^*b^* ✗
$S \to AB$
$A \to aA / \epsilon$
$B \to bB / \epsilon$

$S \to aSb / \cancel{ab} \epsilon$
aaSbb
aaaSbbb

Page No.:
Date:

① $L = \{a^n b^n, n \geq 1\}$

$S \rightarrow aSb \mid ab$

② $L = \{ 0^m 1^m 2^n , m \geq 1, n \geq 0 \}$

$\underbrace{0^m 1^m}_{A} \underbrace{2^n}_{B}$

~~$S \rightarrow 0S1S2$~~

$S \rightarrow AB$

$A \rightarrow 0A1 \mid 01$

$B \rightarrow 2B \mid \epsilon$

③ $L = \{ WW^R : W \in (a+b)^* \}$

$S \rightarrow aSa \mid bSb \mid \epsilon$

a	a
b	b
ab	ba
ba	ab
abb	bba
baa	aab
aba	aba

a .... a S b .... b

↓

C

<u>Balanced Parenthesis</u>

( ) [( )]

[ ]

{ }

one or more pair
hence not ϵ

$S \rightarrow \{S\} \mid (S) \mid [S] \mid ( ) \mid [ ] \mid \{ \}$

⑥ $L = \{ 0^p 1^j, p \neq j, i \& j \geq 0 \}$

$S \rightarrow 0S1 \mid 0S \mid$ ~~ϵ~~ $\mid S1 \mid$

i or j cannot be one together

mostly $S \rightarrow 0S \mid S1 \mid 0 \mid 1$

Page No.:
Date:

(7) $L = \{ a^{n+2} b^m \mid n \geq 0, m > n \}$

~~S~~ $\Rightarrow n = 0, m \geq 1$

minimum one b then b^*

minimum requirement

$n = 0, m \geq 1$:- $\underline{aabb^*}$

(A) (B)

$n = 1, m \geq 2$:- $aaabbb^*$

$n = 2, m \geq 3$:- $a\boxed{aaabbb}b^*$

$S \rightarrow aAB$

$A \rightarrow aAb \mid ab$

$B \rightarrow bB \mid \epsilon$

Topics :

- CFG for Regular Language { F.A → CFG ; R.E → CFG }
- CFG for Non Regular language.

• FA → CFG	• δ (A, a) → S A → a S for final states extra A → ε
• R.E → CFG	• write any * as a variable ie (a+b)* as A (ab)* as B and for the initial string later define A & B's productions
• R.L → CFG	• Thinking !

02/04/18

Page No.:
Date:

- Derivation [Parsing]

Eg

Derivation Tree [Parse Tree]

1. $E \rightarrow E+E$
2. $E \rightarrow E*E$

$E \rightarrow E-E$

$E \rightarrow E/E$

$E \rightarrow (E)/I$

$I \rightarrow$ identity

a) $a+b*c$ valid identifier

RMD

Parsing — Leftmost parsing / Rightmost parsing

RMD → right side variables are expanded first

LMD → left side variables are expanded first

$E \Rightarrow E+E$

$\Rightarrow E+E*E$

$\Rightarrow E+E*I$

$\Rightarrow E+E*id$

$\Rightarrow E+I*id$

$\Rightarrow E+id*id$

$E \Rightarrow I+id*id$

$\Rightarrow id+id*id$

$a+b*c$

terminals.

Eg for rightmost parsing

LMD

$E \Rightarrow E*E$

$E \Rightarrow E+E*E$

$\Rightarrow I+E*E$

$\Rightarrow id+E*E$

$\Rightarrow id+I*E$

$\Rightarrow id+id*E$

$\Rightarrow id+id*I$

$id+id*id$

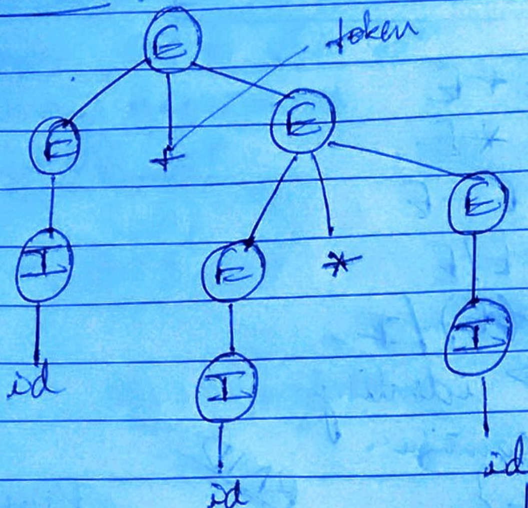
$a+b*c$

Eg for leftmost parsing.

b) $\underline{a + * bc}$ → Invalid string ; Not valid arithmetic operation

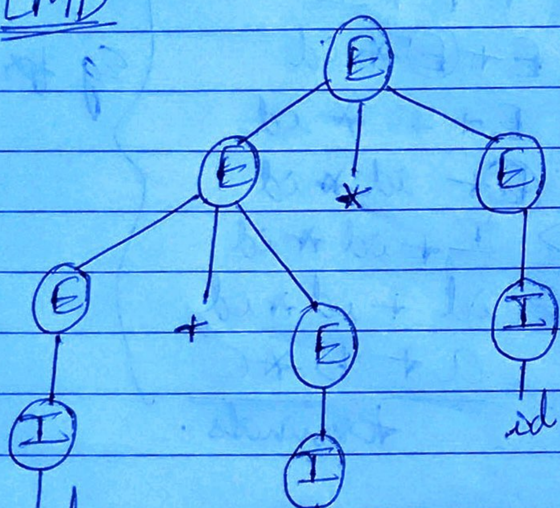
Parse Tree :-

RMD



So analysing the leaf nodes, we get $\boxed{\text{Yield of a parse Tree: } id + id * id}$

LMD



Yield of parse Tree ⇒ $\underline{id + id} * id$

There is a structural difference ⇒
The Given grammer is ambiguous .

Page No.:
Date:

Ambiguous Grammar :- For any given i/p string, if Grammar (G) we can't have :-

i) W . 2 LMDs
2 RMDs
G .

} anyone of these. We get 2 parse tree → P.T structure different then [Ambiguous]

Examples :-

1) S.T the following Grammar is ambiguous.

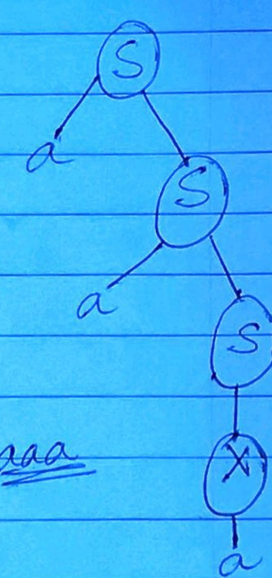
$$S \rightarrow aS \mid X$$
$$X \rightarrow aX \mid a$$

a S
a a X
a a a

W = aaa

① **LMD - 1**

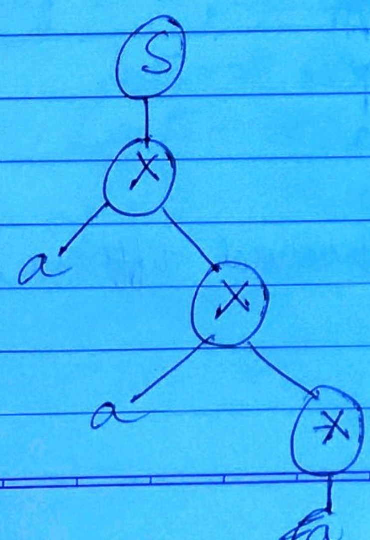
$S \Rightarrow aS$
$\Rightarrow aaS$ // Rule $S \rightarrow aS$
$\Rightarrow aa\cancel{X}$ // $S \rightarrow X$
$\Rightarrow aaa$ // $X \rightarrow a$



aaa

LMD - 2

$S \Rightarrow X$
$\Rightarrow aX$ // $X \rightarrow aX$
$\Rightarrow aaX$ // $X \rightarrow aX$
$\Rightarrow aaa$ // $X \rightarrow a$

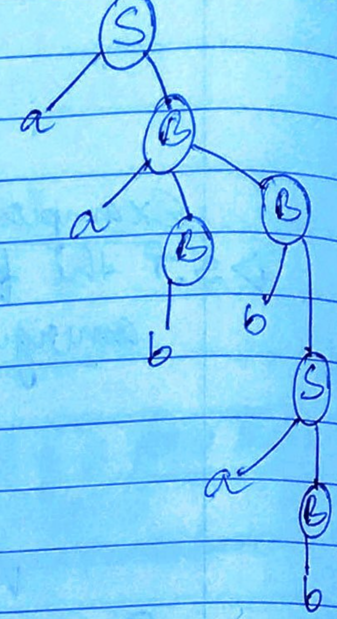


aaa

Structural difference
⇓
ambiguous.

2) $W = aabbab$

$$S \rightarrow aB \mid bA$$
$$A \rightarrow aS \mid bAA \mid a$$
$$B \rightarrow bS \mid aBB \mid b$$



LMD-1

$S \Rightarrow aB\cancel{bA}$

$\cancel{B} \Rightarrow aaBB$ // $B \rightarrow aBB$

$\Rightarrow aabB$ // $B \rightarrow b$

$\Rightarrow aabbS$ // $B \rightarrow bS$

$\Rightarrow aabbaB$ // $S \rightarrow aB$

$\Rightarrow aabbab$ // $B \rightarrow b$

LMD-2

$S \Rightarrow aB$

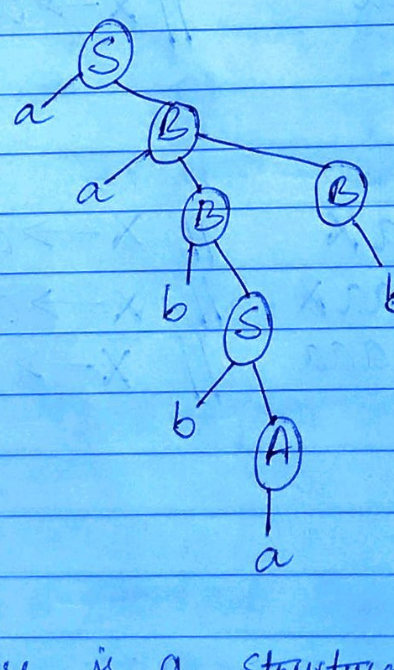
$\Rightarrow aaBB$ // $B \rightarrow aBB$

$\Rightarrow aabSB$ // $B \rightarrow bS$

$\Rightarrow aabbAB$ // $S \rightarrow bA$

$\Rightarrow aabbaB$ // $A \rightarrow a$

$\Rightarrow aabbab$ // $B \rightarrow b$



As there is a structural difference, the parse tree is ambiguous.

Representation ① $S \overset{*}{\underset{rm}{\Longrightarrow}} W$ — RMD

any no of derivations

② $S \overset{*}{\underset{lw}{\Longrightarrow}} W$ — LMD

Languages :-

DFA

$L = \{ W \mid \hat{\delta}(q_0, W) \in F \}$

NFA

$CFG = L = \{ W \mid S \overset{*}{\Longrightarrow} W \}$

if any one derivation starting from S results in string w.

* Normal Forms :-

① Chomsky NF (CNF)

② Greibach NF (GNF)

10-12 Marks

① Chomsky NF :- CNF

- $A \longrightarrow BC$
- $A \longrightarrow a$

$$\boxed{\begin{array}{l} E \rightarrow EE \\ E \rightarrow a \end{array}}$$

a) Eliminating ϵ - production. $[\cancel{A \rightarrow \epsilon}]$

b) Eliminating unit - production $[\cancel{A \rightarrow B}]$ single variable b/w terminal allowed

c) Eliminating Useless Symbols.

1) Eliminating ϵ - production :-

Rules

1> $S \longrightarrow ABC$

2> $A \longrightarrow BC \mid a$

3> $B \longrightarrow bAC \mid \epsilon$

4> $C \longrightarrow cAB \mid \epsilon$

Page No.:
Date:

a) Identify the Nullable symbol :-
↳ any symbol which derives ϵ.

$NS = \{B, C\}$ $A \to \epsilon$

$B \to \epsilon$
$C \to \epsilon$

i) Step i) ~~A is $\to \epsilon$~~, ~~A is NS~~
i) $V \to \epsilon$, A is NS
ii) $V \to V_1, V_2, \ldots V_n$, A is NS
if $V_1 V_2 \ldots V_n$ are NS

• if $A \to BaC$ was there we wont take.

$A \to BC | a$ [Since B, C are list, put A in the list]

$NS = \{A, B, C\}$

• ie if terminal is there we wont take

$NS = \{A, B, C, S\}$ as $S \to ABC$

Only variables
* If there are any terminals in the set, don't consider.

$NS = \{\overbrace{B, C}^{i}, \overbrace{A, S}^{ii}\}$

New rules free from ϵ production

i) $S \to ABC | BC | AC | AB | C | A | B$
$A \to \epsilon$, $B \to \epsilon$, $C \to \epsilon$, $AB \to \epsilon$, $B \to \epsilon$, $AC \to \epsilon$

ii) $A \to BC | C | B | a$
iii) $B \to bAC | bc | bA | b$
iv) $C \to cAB | cA | cB | c$

2) Eliminate ϵ-productions

i) $S \to ABC | BaB$
ii) $A \to aA | BaC | aaa$
iii) $B \to bBb | a | D$
iv) $C \to CA | AC$
v) $D \to \epsilon$

Page No.:	
Date:	

a) Identify Nullable Symbol (NS)

$NS = \{\overset{i}{D}, \overset{ii}{B}\}$

i) $A \to \epsilon$

ii) $A \to \underbrace{V_1 V_2 \ldots V_n}_{NS}$

ε free production

i) $S \to ABC \mid \overset{B\to\epsilon}{AC} \mid \overset{B\to\epsilon}{BaB} \mid Ba \mid a$

ii) $A \to aA \mid BaG \mid aG \mid aaa$

iii) $B \to bBb \mid bb \mid a \mid D$

iv) $C \to CA \mid AC$

2) $S \to AAA \mid B$

$A \to aA \mid B$

$B \to \epsilon$

$NS = \{\overset{i}{B}, \overset{ii}{A}, S\}$

i) ~~$S \to AAA \mid AA \mid AA \mid B\,AA \mid A \mid A$~~

ii) ~~$A \to aA \mid B \mid a$~~

~~iii)~~

i) $S \to AAA \mid AA \mid A \mid B$

ii) $A \to aA \mid B \mid a$

4) $S \to 0A0 \mid 1B1 \mid BB$

$A \to G$

$B \to S \mid A$

$G \to S \mid \epsilon$

$NS = \{\overset{i}{G}, \overset{ii}{A}, B, S\}$

i) $S \to 0A0 \mid 00 \mid 1B1 \mid 11 \mid BB \mid B$

ii) $A \to G$

iii) $B \to S \mid A$

iv) $C \to S$

2 step :- Eliminating Unit productions :- $A \to B$

- i) (X, X)
- ii) (X, Y)

1)

	Unit pair	Productions
i)	(E, E)	$E \to E+T$
	(T, T)	$T \to T*F$
	(F, F)	$F \to (E)$
	(I, I)	$I \to a \mid b \mid Ia \mid Ib \mid I0 \mid I1$
ii)	(E, T)	$E \to T*F$
	(E, F)	$E \to (E)$
	(E, I)	$E \to a \mid b \mid Ia \mid Ib \mid I0 \mid I1$
	(T, F)	$T \to (E)$
	(T, I)	$T \to a \mid b \mid Ia \mid Ib \mid I0 \mid I1$
	(F, I)	$F \to a \mid b \mid Ia \mid Ib \mid I0 \mid I1$

1st, 2nd

Unit production
$X \to \cdots \to Y$

Assign Non-unit production of 2nd variable to first (1st) variable

1) $E \to E+T \mid T$
$T \to T*F \mid F$
$F \to (E) \mid I$
$I \to a \mid b \mid Ia \mid Ib \mid I0 \mid I1$

$$E \to T \to F \to I$$

$$T \to F \to I$$

$$F \to I$$

—— " ——

Page No.:
Date:

2) $S \rightarrow AAA \mid B$
$A \rightarrow aA \mid B$.
$B \rightarrow b$

	Unit pair	Production
i)	(S, S)	$S \rightarrow AAA$
	(A, A)	$A \rightarrow aA$
	(B, B)	$B \rightarrow b$
ii)	(S, B)	$S \rightarrow b$
	(A, B)	$A \rightarrow b$

3) $S \rightarrow 0A0 \mid 1B1 \mid BB$
$A \rightarrow C$
$B \rightarrow S \mid A$
$C \rightarrow S \mid \cancel{A} \mid C$

	Unit pair	Production
i)	(S, S)	$S \rightarrow 0A0 \mid 1B1 \mid BB$
	(A, A)	$A \rightarrow \cancel{C} S$ = ? no unit production
	(B, B)	$B \rightarrow S$
	(C, C)	$C \rightarrow S$ = ?
ii	(A, C)	$\cancel{A \rightarrow S}$
	(A, S)	$A \rightarrow 0A0 \mid 1B1 \mid BB$
	(C, S)	$C \rightarrow S$
	(B, S)	$B \rightarrow 0A0 \mid 1B1 \mid BB$?
	(B, A)	$B \rightarrow S$

03/04/18

Page No.:
Date:

$S \to A/c$
$A \to B$
here A is Non Generating symbol.

3) Eliminating Useless Symbols :-

then
a) Non - Generating symbols [Symbols that does'nt generate any terminal]

b) Non - Reachable symbols [Any symbol Not reachable]

$S \to aA/b$
$C \to A$
then C is not reachable for starting state [S]

a) Non Generating Symbols (NGS) :-

$GS = \{S\}$
$NGS = \{\{V, T\} - GS\}$

Generating Symbols :- Symbols which generate terminals.

Steps :- i) Terminals are Generating symbols
* ii) $A \to \alpha$, all symbols in α are generating then A is a GS.

Eg :- i) $S \to AB/AC$
$A \to aA/bAa/a$
$B \to bbA/aB/AB$
$C \to aCa/aD$
$D \to aD/bC$

$GS = \{\overbrace{a, b}^{(i)}, \overset{(ii)}{A}, \overset{(iii)}{B}, \overset{(iv)}{S}\}$
$\Rightarrow$
So the $NGS = \{C, D\}$

$(V \cup T)^*$
$A \to \alpha$
$\downarrow$
a

Page No.:
Date:

So now, wherever there are C and D in the production, we remove them from the Grammer.

$$\Rightarrow S \to AB \mid \cancel{AC}$$ eliminated

$$A \to aA \mid bAa \mid a$$

$$B \to bbA \mid aB \mid AB$$

Eliminated: $\cancel{C \to aCa \mid aD}$, $\cancel{D \to aD \mid bC}$

(2) Eliminate the NGS from the given grammer.

$$S \to aAa \mid bBb$$
$$A \to a$$
$$B \to b$$
$$C \to CDE$$
$$D \to ab$$

$$GS = \{\overbrace{a, b}^{(i)}, A, B, S, D\}$$

$$NGS = \cancel{\{D\}}\{C, E\}$$

Eliminating C & E from the productions.

$$\Rightarrow S \to aAa \mid bBb$$
$$A \to a$$
$$B \to b$$
$$D \to ab$$

Eliminated Useless Symbols.

~~(1) $S \to AB$~~
~~$A \to aA \mid bAa \mid a$~~
~~$B \to bbA \mid aB \mid AB$~~
~~$GS = \{a, b, A, B, S\}$~~

Page No.:
Date:

(2) $S \rightarrow ABC \mid BaB$
$A \rightarrow aA \mid BaC \mid aaa$
$B \rightarrow bBb \mid a$
$C \rightarrow CA \mid AC$

$\Rightarrow GS = \{a, b, A, B, S\}$
$NGS = \{C\}$
Eliminate C from production
$\Rightarrow S \rightarrow BaB$
$A \rightarrow aA \mid aaa$
$B \rightarrow bBb \mid a$

b) Reachable Symbols :-
i) S is reachable
ii) $A \rightarrow \alpha$, A is RS, then all symbols in α are RS

After eliminating NGS,
1) $S \rightarrow AB$ α
$A \rightarrow aA \mid bAa \mid a$
$B \rightarrow bbA \mid aB \mid AB$

$RS = \{S, A, B, a, b\}$
$NRS = \{Nil\}$
So free from NRS

if NRS contains terminal will we ever remove it?

2) $S \rightarrow aAa \mid bBb$
$A \rightarrow a$
$B \rightarrow b$
$D \rightarrow ab$

$RS = \{S, a, b, A, B\}$
$NRS = \{D\}$
$\Rightarrow$ Production free from D $\Rightarrow$

Page No.:
Date:

$$\left\{\begin{array}{l} S \to aAa \mid bBb \\ A \to a \\ B \to b \end{array}\right.$$ ✓

3) $S \to BaB$
$A \to aA \mid aaa$
$B \to bBb \mid a$

$RS = \{S, B, a\}$
$NRS = \{A\}$

Eliminate NRS from production.

$$\Rightarrow \left\{\begin{array}{l} S \to BaB \\ B \to bBb \mid a \end{array}\right.$$ ✓

~~After~~ these 3 steps apply one more step to convert to Chomsky Normal form :-

$S \to AB$ ✓
$A \to aA \mid bAa \mid a$ ✓
$B \to bbA \mid aB \mid AB$ ✓

CNF

(4) $E \to E + T \mid a$ ✓
$T \to T * F \mid b$ ✓
$F \to (E) \mid a \mid b$ ✓

P, M, O, C

i) $E \to EPT \mid a$ — Three variables Not allowed
$T \to TMF \mid b$

$F \to OEC \mid a \mid b$

$P \to +$
$M \to *$
$O \to ($, $C \to)$

ii) $E \rightarrow QT \mid a$

$T \rightarrow RF \mid b$

$F \rightarrow WG \mid a \mid b$

$Q \rightarrow EP$

$R \rightarrow TM$

$W \rightarrow OE$

Grammer is in Chomsky Normal Form

$P \rightarrow +$

$M \rightarrow *$

$O \rightarrow ($, $G \rightarrow)$

2) $S \rightarrow AB$

$A \rightarrow aA \mid bAa \mid a$

$B \rightarrow bbA \mid aB \mid AB$

(Step 1)

a) i) $S \rightarrow AB$

$A \rightarrow XA \mid YAX \mid a$

$B \rightarrow YYA \mid XB \mid AB$

$X \rightarrow a$

$Y \rightarrow b$

Step 2

$S \rightarrow AB$

$A \rightarrow XA \mid WX \mid a$

$B \rightarrow ZA \mid XB \mid AB$

$X \rightarrow a$

$Y \rightarrow b$

$W \rightarrow YA$

$Z \rightarrow YY$

Chomsky Normal Form

Page No.:
Date:

1) Consider the following grammer :-

$S \rightarrow ABC \mid BaB$
$A \rightarrow aA \mid BaC \mid aaa$
$B \rightarrow bBb \mid a \mid D$
$C \rightarrow CA \mid AC$
$D \rightarrow \epsilon$

a) Eliminate ϵ productions
b) Eliminate unit productions in resulting Grammer
c) —— " useless symbols —— " ——
d) Put resulting symbol into Chomsky NF

a) $NS = \{D, B\}$
$S \rightarrow ABC \mid AC \mid BaB \mid aB \mid Ba \mid a$
$A \rightarrow aA \mid BaC \mid aC \mid aaa$
$B \rightarrow bBb \mid bb \mid a \mid D$
$C \rightarrow CA \mid AC$

b)

Unit pair	Production
i) (S, S)	$S \rightarrow ABC \mid AC \mid BaB \mid aB \mid Ba \mid a$
(A, A)	$A \rightarrow aA \mid BaC \mid aC \mid aaa$
(B, B)	$B \rightarrow bBb \mid bb \mid a$
(C, C)	$C \rightarrow CA \mid AC$
(B, D)	

c) $S \rightarrow ABC \mid AC \mid BaB \mid aB \mid Ba \mid a$
$A \rightarrow aA \mid BaC \mid aC \mid aaa$
$B \rightarrow bBb \mid bb \mid a$
$C \rightarrow CA \mid AC$

$GS = \{a, b, B, S, A\}$
$NGS = \{C\}$

Page No.:

Date:

eliminate C

$S \to BaB \mid aB \mid Ba \mid a$

$A \to aA \mid aaa$

$B \to bBb \mid bb \mid a$

$RS = \{S, B, a, b\}$

$NRS = \{A\}$

$\left\{ \begin{array}{l} S \to BaB \mid aB \mid Ba \mid a \\ B \to bBb \mid bb \mid a \end{array} \right.$

iv) i) $S \to BXB \mid XB \mid BX \mid a$

$B \to YBY \mid YY \mid a$

$X \to a$

$Y \to b$

Chomsky Normal Form

ii) $S \to ZB \mid XB \mid BX \mid a$

$B \to QY \mid YY \mid a$

$X \to a$

$Y \to b$

$Z \to BX$

$Q \to YB$

---

2) $S \to AAA \mid B$

$A \to aA \mid B$

$B \to \epsilon$

a) $NS = \{B, A, S\}$

$\Rightarrow \left\{ \begin{array}{l} S \to AAA \mid AA \mid A \mid B \\ A \to aA \mid a \mid B \end{array} \right.$

Page No.:
Date:

(b) Unit Production

(S, S)	$S \to AAA \mid AA \mid \cancel{A}$
(A, A)	$A \to aA \mid a$
(A, B)	$A \to$ ______
(S, B)	$S \to$ ______
(S, A)	$S \to aA \mid a$

$\Rightarrow$

(c) $S \to AAA \mid AA \cancel{\mid}$
$A \to aA \mid a$
$S \to aA \mid a$

GS = {a, A, S}
NGS = {Nil}

ii) RS = {S, A, a}
NRS = {Nil}

d) $S \to AAA \mid AA$
$A \to aA \mid a$
$S \to aA \mid a$

i) $S \to AAA \mid AA$
$A \to XA \mid a$
$S \to XA \mid a$

$X \to a$

ii) $S \to PA \mid AA$
$A \to XA \mid a$
$S \to XA \mid a$ ✓
$X \to a$
$P \to AA$

Chomsky Normal Form

Page No.:
Date:

II Greibach Normal form (GNF) :-

non GNF to GNF.

$S \rightarrow AA \mid 0$
$A \rightarrow SS \mid 1$

1) CNF
2) Rename variables as $A_1, A_2 \ldots$

~~A~~$_1$

(i) It is CNF
(ii) $A_1 \rightarrow A_2 A_2 \mid 0 \longrightarrow$ ①
$A_2 \rightarrow A_1 A_1 \mid 1 \longrightarrow$ ②

3) $A_i \longrightarrow A_j \alpha \quad , i < j$
$i = 1, j = 2 \quad i < j$

Consider A_2 - production.
$A_2 \rightarrow A_1 A_1 \mid 1$
Substitute A_1 - production.
$A_2 \rightarrow A_2 A_2 A_1 \mid 0 A_1 \mid 1$ must be same
Left recursive producⁿ $\Rightarrow$ $A \rightarrow AB$

$$\boxed{A \longrightarrow A\alpha \mid B \quad \equiv \quad \begin{array}{l} A \rightarrow BZ \mid B \\ Z \rightarrow \alpha Z \mid \alpha \end{array}}$$ $\mid \beta_2 Z \mid \beta_2$

After eliminating left-recursive production

③ $\Rightarrow$ $A_2 \longrightarrow \overset{\beta_1 Z}{0 A_1 Z} \mid \overset{\beta_2 Z}{1 Z} \mid \overset{\beta_1}{0 A_1} \mid \overset{\beta_2}{1}$
$Z \longrightarrow \underset{\alpha Z}{A_2 A_1 Z} \mid \underset{\alpha}{A_2 A_1}$

④ Consider Z producⁿ.
Substitute A_2 producⁿ.

⑤ $\Rightarrow$

$Z \rightarrow 0 A_1 Z A_1 Z \mid 1 Z A_1 Z \mid 0 A_1 A_1 Z \mid 1 A_1 Z \mid 0 A_1 Z A_1 \mid 1 Z A_1 \mid 0 A_1 A_1 \mid$ 1...

Page No.:
Date:

substituting A_2 production in A_1

$A_1 \longrightarrow 0A_1ZA_2 \mid ZA_2 \mid 0A_1A_2 \mid 1A_2 \mid 0$

$A_2 \rightarrow 0A_1Z \mid 1Z \mid 0A_1 \mid 1$
$A_1 \rightarrow 0A_1 ZA_2 \mid 1ZA_2 \mid 0A_1A_2 \mid 1A_2 \mid 0$.
$Z \rightarrow$ ......

Convert the following ~~to~~ non GNF to GNF.

Q) $A \rightarrow BC$
$B \rightarrow CA \mid b$
$C \rightarrow AB \mid a$

1) It is in CNF.
2) $A_1 \rightarrow A_2A_3$
$A_2 \rightarrow A_3A_1 \mid b$.
$A_3 \rightarrow A_1A_2 \mid a$.

3) $A_i \rightarrow A_j\,\alpha$, ~~i < j~~ $i > j$.
$A_3 \longrightarrow A_1A_2 \mid a$.
$A_1 \rightarrow A_2A_3$.
$A_3 \longrightarrow A_2A_3A_2 \mid a$
$A_3 \rightarrow \cancel{A_2A_1}\, A_3A_1A_3A_2 \mid b\,\overset{B_1}{A_3}\overset{B_2}{A_2} \mid a$.

after eliminating left recursive producn.
$A_3 \longrightarrow bA_3A_2Z \mid aZ \mid bA_3A_2 \mid a$.
$Z \longrightarrow A_1A_3A_2Z \mid A_1A_3A_2$.

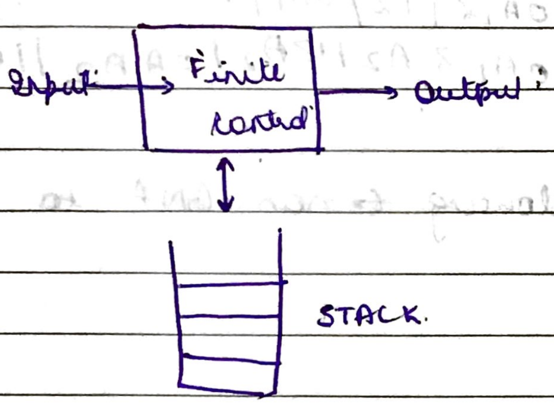
~~consider Z producn~~ consider A_2 production
~~Sub A_1~~ $A_2 \rightarrow bA_3A_2ZA_1 \mid aZA_1 \mid bA_3A_2A_1 \mid aA_1 \mid b$ (5).

$Z \rightarrow A_2A_3$
A_2 producn in GNF.

* if in a state q_1 we push then never keep an pop operation in the state state or vice versa

Unit -4

Push Down Automata (PDA)



$(Q, \Sigma, \Gamma, \delta, q_0, z_0, F)$

Q : Set of states

Γ : Set of STACK symbols.

δ : Transmission func^{n}s.

q_0 : starting state

z_0 : Initial content of STACK.

F : Set of Final states.

PUSH :- current state, i/p, top, moving, PUSH

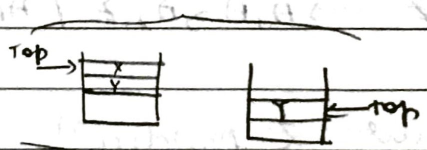
$$\delta(q_0, a, x) = (q_1, ax)$$

POP

$$\delta(q_0, a, x) = (q_1, \epsilon)$$

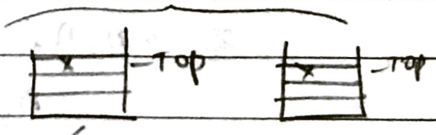
No-operan.

$$\delta(q_0, a, x) = (q_1, \cancel{a} x)$$



Replace

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



Page No.:
Date:

} Replace

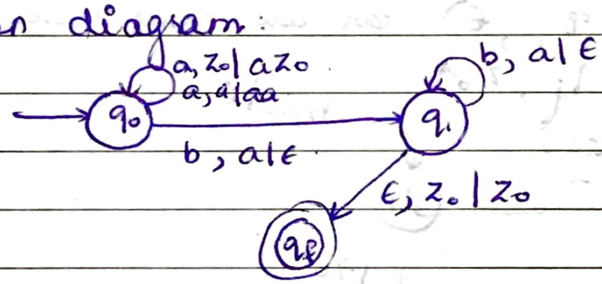
1) $\delta(P, a, z) = (q, a z)$ - Push $= (q, a_a z)$ - Push
2) $\delta(P, a, z) = (P, \epsilon)$ - Pop
3) $\delta(P, a, z) = (q, \wedge)$ - Replace
4) $\delta(q, \epsilon, z) = (q_1, \epsilon)$ - ~~No~~ Pop
5) $\delta(q, a, Z) = (q, Y)$ - Replace

1) $a^n b^n$ $n \geq 1$, for $n \geq 0$, $\delta(q_0, \epsilon, z_0) = (q_f, z_0)$.

a) $\delta(q_0, \overset{i/p}{a}, z_0) = (q_0, \overset{pushed}{a} z_0)$
b) $(q_0, a, a) = (q_0, aa)$
c) $\delta(q_0, b, a) = (q_1, \epsilon)$
d) ~~$\delta(q_0, \epsilon, z_0) = (q_f, Z_0)$ or (q_f, ϵ)~~
d) $\delta(q_1, b, a) = (q_1, \epsilon)$
e) $\delta(q_1, \epsilon, z_0) = (q_f, z_0)$ or (q_f, ϵ)

1 Method

Transition diagram:



(i) w = aabb, i/p string
$\delta(q_0, aabb, z_0)$
$\vdash \hat{\delta}(q_0, abb, az_0)$
$\vdash \hat{\delta}(q_0, bb, aaz_0)$
$\vdash \hat{\delta}(q_1, b, az_0)$
$\vdash \hat{\delta}(q_1, \epsilon, z_0) \vdash (q_f, \epsilon, z_0)$ $\longrightarrow$ (Reached final state q_f, so i/p string is valid)

(ii) $w = abb$

$\hat{\delta}(q_0, abb, Z_0)$

$\vdash \hat{\delta}(q_0, bb, aZ_0)$

$\vdash \hat{\delta}(q_1, b, Z_0)$

Transiⁿ is not existing so invalid

(iii) $w = aab$

$\hat{\delta}(q_0, aab, Z_0)$

$\vdash \hat{\delta}(q_0, ab, aZ_0)$

$\vdash \hat{\delta}(q_0, b, aaZ_0)$

$\vdash \hat{\delta}(q_1, \epsilon, aZ_0)$

Transiⁿ is not existing so invalid.

(2) $a^n b^{2n}$, $n \geq 1$, Construct a PDA.

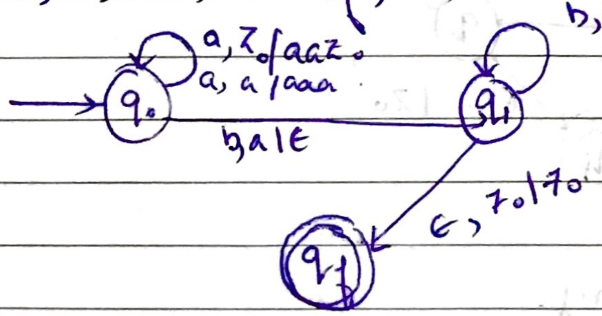
$\hat{\delta}(q_0, a, Z_0) = (q_0, aaZ_0)$

$\delta(q_0, a, a) = (q_0, \boxed{aa}a)$ for next o 2nd transiⁿ can be repeated so enough

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

$\delta(q_1, b, a) = (q_1, \epsilon)$ can use same transiⁿs to pop b's.

$\delta(q_1, \epsilon, Z_0) = (q_f, Z_0)$



i/p string $w = aabbbb$

$\hat{\delta}(q_0, aabbbb, Z_0)$

$\vdash \hat{\delta}(q_0, abbbb, aaZ_0)$

$\vdash \hat{\delta}(q_0, bbbb, aaaaZ_0)$

$\vdash \hat{\delta}(q_1, bbb, aaaZ_0)$

$\vdash \hat{\delta}(q_1, bb, aaZ_0)$

$\vdash \hat{\delta}(q_1, b, aZ_0)$

$\vdash \hat{\delta}(q_1, \epsilon, Z_0)$

$\vdash (q_f, Z_0) \rightarrow$ valid.

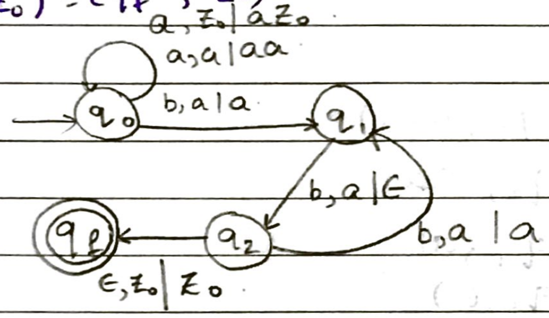
2) $w = aab$.

$\hat{\delta}(q_0, aab, z_0) \vdash \hat{\delta}(q_0, ab, aaz_0) \vdash$
$\hat{\delta}(q_0, b, aaaaz_0) \vdash \hat{\delta}(q_1, \epsilon, aaaz_0)$.

Invalid.

<u>Second solution</u>:

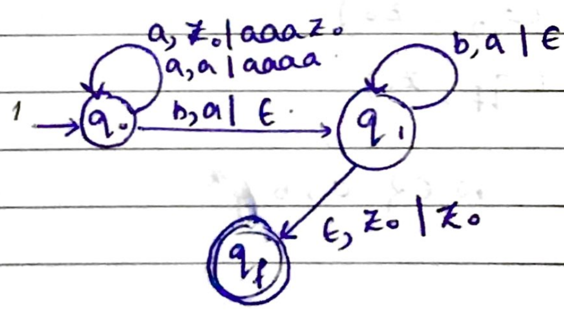
① $\delta(q_0, a, z_0) = (q_0, az_0)$
② $\delta(q_0, a, a) = (q_0, aa)$
③ $\delta(q_0, b, a) = (q_1, a)$
④ $\delta(q_1, b, a) = (q_2, \epsilon)$
⑤ $\delta(q_2, b, a) = (q_1, a)$
⑥ $\delta(q_2, \epsilon, z_0) = (q_f, z_0)$



Q) $a^n b^{3n}$

aabbbbbb

① $\delta(q_0, a, z_0) = (q_0, aaaz_0)$
② $\delta(q_0, a, a) = (q_0, aaaa)$
③ $\delta(q_0, b, a) = (q_1, \epsilon)$
④ $\delta(q_1, b, a) = (q_1, \epsilon)$
⑤ $\delta(q_1, \epsilon, z_0) = (q_f, z_0)$



4) $a^n b^{n+1}$, $n \geq 1$ aabbb

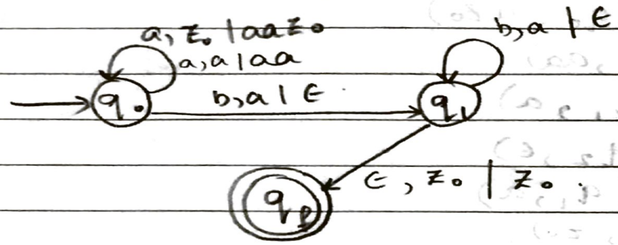
① $\delta(q_0, a, z_0) = (q_0, aaz_0)$.

② $\delta(q_0, a, a) = (q_0, aa)$

③ $\delta(q_0, b, a) = (q_1, \epsilon)$

④ $\delta(q_1, b, a) = (q_1, \epsilon)$

⑤ $\delta(q_1, \epsilon, z_0) = (q_f, z_0)$.



Second soln!

① $\delta(q_0, a, z_0) = (q_0, az_0)$

② $\delta(q_0, a, a) = (q_0, aa)$.

//No operan ③ $\delta(q_0, b, a) = (q_1, a)$

④ $\delta(q_1, b, a) = (q_1, \epsilon)$

⑤ $\delta(q_1, \epsilon, z_0) = (q_f, z_0)$.

⑥

Third soln

① $\delta(q_0, a, z_0) = (q_0, az_0)$

② $\delta(q_0, a, a) = (q_0, aa)$

③ $\delta(q_0, b, a) = (q_1, \epsilon)$

④ $\delta(q_1, b, a) = (q_1, \epsilon)$

⑤ $\delta(q_1, b, z_0) = (q_2, z_0)$ // there may be extra b's //

⑥ $\delta(q_2, \epsilon, z_0) = (q_f, z_0)$

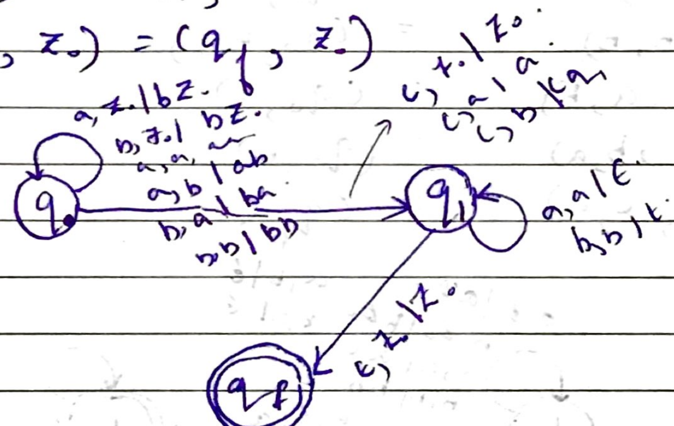
5) $w c w^R$ $w \in \{a, b\}^*$. , a b c b a.

Page No.:
Date:

Q) Design a PDA which accepts the language.

$L = \{ w c w^R \mid w \in (a+b)^* \}$.

1) $\delta(q_0, a, z_0) = (q_0, a z_0)$. // 1st char 'a'
2) $\delta(q_0, b, z_0) = (q_0, b z_0)$. // — " — 'b'
3) $\delta(q_0, a, a) = (q_0, aa)$
4) $\delta(q_0, a, b) = (q_0, ab)$
5) $\delta(q_0, b, a) = (q_0, ba)$
6) $\delta(q_0, b, b) = (q_0, bb)$.
7) $\delta(q_0, c, z_0) = (q_1, z_0)$
8) $\delta(q_0, c, a) = (q_1, a)$
9) $\delta(q_0, c, b) = (q_1, b)$.
10) $\delta(q_1, a, a) = (q_1, \epsilon)$
11) $\delta(q_1, b, b) = (q_1, \epsilon)$.
12) $\delta(q_1, \epsilon, z_0) = (q_f, z_0)$



Q. Let $= \{ w w^R \mid w \in (a+b)^* \}$. (WWR)

q_0

abba.

~~$(q_0, abba, z_0)$~~

abba :

$(q_0, abba, Z_0)$

$\vdash (q_0, bbb, aZ_0)$

$\vdash (q_0, ba, baZ_0)$

$\cancel{\vdash (q_1, a, aZ_0)}$

$\cancel{\vdash \rightarrow q_0 \; q_f \; \epsilon, Z_0}$

Design a PDA to accept language :-

1) $L = \{ w \mid w \in (a+b)^* \; , n_a(w) > n_b(w) \}$. $\quad a^n b^n$

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

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

① $\delta(q_0, a, Z_0) = (q_0, aZ_0)$

② $\delta(q_0, b, Z_0) = (q_0, bZ_0)$

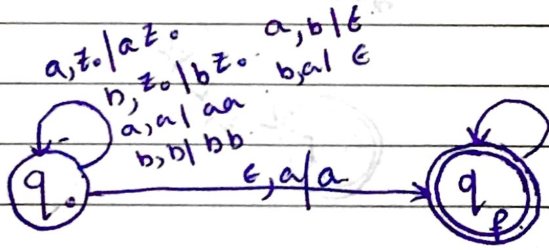
③ $\delta(q_0, a, a) = (q_0, aa)$

④ $\delta(q_0, b, b) = (q_0, bb)$

⑤ $\delta(q_0, a, b) = (q_0, \epsilon)$

6) $\delta(q_0, b, a) = (q_0, \epsilon)$

7) $\delta(q_0, \epsilon, a) = (q_f, a)$



Page No.:
Date:

Language of PDA

1) Acceptance by final states

$$L = \{ w \mid (q_0, w, z_0) \vdash^* (q_f, \epsilon, \alpha) \}$$

2) Acceptance by empty stack :-

$$L = \{ w \mid (q_0, w, z_0) \vdash^* (q, \epsilon, \epsilon) \}$$

After all the transitions stack must be empty

Types of PDA

1) Deterministic PDA (DPDA)
2) Non-Deterministic PDA (NPDA).

Condi^ns for DPDA.

1) $\delta(q, a, X)$ must have only one definition.

2) $\delta(q, \epsilon, X)$ is non-empty then $\delta(q, a, X)$ should be empty

Q) Is the PDA to accept the language;

$$L = \{ w \mid w \in (a+b)^* \quad n_a(w) > n_b(w) \}$$ is deterministic ?

1) $\delta(q_0, a, z_0) = (q_0, az_0)$
2) $\delta(q_0, b, z_0) = (q_0, bz_0)$
3) $\delta(q_0, a, a) = (q_0, aa)$
4) $\delta(q_0, b, b) = (q_0, bb)$
5) $\delta(q_0, a, b) = (q_0, \epsilon)$
6) $\delta(q_0, b, a) = (q_0, \epsilon)$
7) $\delta(q_0, \epsilon, a) = (q_f, a)$

It is not deterministic, $\therefore$ it is NPDA.

Page No.:
Date:

2) wCw^R (DPDA)
3) ww^R (NPDA)

$\delta(q_0, \epsilon, Z_0) \rightarrow (q_1, SZ_0)$
$\delta(q_1, a, S) \rightarrow (q_1, ABC)$
$\delta(q_1, a, A) \rightarrow (q_1, B)$
$\delta(q_1, a, A) \rightarrow (q_1, \epsilon)$

CFG to PDA

1) GNF
2) $\delta(q_0, \epsilon, Z_0) = (q_1, SZ_0)$
3) $A \rightarrow b\alpha$, $\delta(q_1, b, A) = (q_1, \alpha)$
4) $\delta(q_1, \epsilon, Z_0) = (q_f, Z_0)$

Obtain the PDA for the given grammar :-

I) $S \rightarrow aABC$
$A \rightarrow aB \mid a$
$B \rightarrow bA \mid b$
$C \rightarrow a$

① given CFG is in GNF
2) $\delta(q_0, \epsilon, Z_0) = (q_1, SZ_0)$
3) (i) $S \rightarrow a\underbrace{ABC}_{\alpha}$, $\delta(q_1, a, S) = (q_1, ABC)$
(ii) $A \rightarrow aB$, $\delta(q_1, a, A) = (q_1, B)$
$A \rightarrow a$, $\delta(q_1, a, A) = (q_1, \epsilon)$
(iii) $B \rightarrow bA$, $\delta(q_1, b, B) = (q_1, A)$
$B \rightarrow b$, $\delta(q_1, b, B) = (q_1, \epsilon)$
(iv) $C \rightarrow a$, $\delta(q_1, a, C) = (q_1, \epsilon)$
4) $\delta(q_1, \epsilon, Z_0) = (q_f, Z_0)$

~~① w = aabb~~
~~$(q_0, aaba, Z_0) \vdash (q_1, aaba, SZ_0) \vdash (q_1, aba, ABCZ_0) \vdash$~~
~~$(q_1, ba, BBCZ_0) \vdash (q_1, a, ABCZ_0) \vdash (q_1, \epsilon, ABCZ_0) \vdash$~~

$w = aaba$.

$(q_0, aaba, Z_0) \vdash (q_1, aaba, SZ_0) \vdash (q_1, aba, ABCZ_0) \vdash$
$(q_1, ba, BCZ_0) \vdash (q_1, a, CZ_0) \vdash (q_1, \epsilon, Z_0) \vdash (q_f, \epsilon, Z_0)$

$\therefore$ valid

Obtain the PDA.

1.) $S \rightarrow \cancel{aBAA}$ $aABB \mid aAA$.

$A \rightarrow aBB \mid a$.

$B \rightarrow bBB \mid A \equiv B \rightarrow bBB \mid aBB \mid a$.

$C \rightarrow a$.

1.) given CFG is in GNF.

1) w = aaba

$(q_0, aaba, Z_0) \vdash (q_1, aaba, SZ_0)$ (2) 3 i (b)

$\vdash (q_1, aba, ABCZ_0)$ 3 ii b $\vdash (q_1, ba, BBCZ_0)$ 3 iii b

$\vdash (q_1, a, \cancel{AB}CZ_0)$ iv $\vdash (q_1, \epsilon, Z_0) \vdash (q_f, Z_0)$ (4)

6) i) $S \to aABB / aAA$

ii) $A \to aBB / a$

iii) $B \to bBB / A$

iv) $C \to a$

It is not in GNF convert to GNF

$S \to aABB / aAA$

$A \to aBB / a$

$B \to bBB / aBB / a$

$C \to a$

1) Given CFG is in GNF

2) $\delta(q_0, \epsilon, Z_0) = (q_1, SZ_0)$

3) i) $S \to aABB$, $\delta(q_1, a, S) = (q_1, ABB)$

$S \to aAA$, $\delta(q_1, a, S) = (q_1, AA)$

ii) $A \to aBB$, $\delta(q_1, a, A) = (q_1, BB)$

$A \to a$, $\delta(q_1, a, A) = (q_1, \epsilon)$

iii) $B \to bBB$, $\delta(q_1, b, B) = (q_1, BB)$

$B \to aBB$, $\delta(q_1, a, B) = (q_1, BB)$

$B \to a$, $\delta(q_1, a, B) = (q_1, \epsilon)$

iv) $C \to a$, $\delta(q_1, a, C) = (q_1, \epsilon)$

Date / /20

4) $\delta(q_1, \epsilon, z_0) \Rightarrow (q_f, z_0)$

II PDA to CFG

i) $\delta(q_i, a, Z) = (q_j, AB)$

Productions:- [q_k & q_L are all possible state of Q]

$\underbrace{(q_i Z q_k)}_{\text{one variable}} \rightarrow a \underbrace{(q_j A q_l)}_{\text{one variable}} (q_l B q_k) \rightarrow$ one variable

ii) $\delta(q_i, a, z) = (q_j, \epsilon)$

Production : $q_i Z q_j \rightarrow a$

Q) $\delta(q_0, a, z) = (q_0, AZ)$ — ⓐ ① method

$\delta(q_0, b, A) = (q_0, AA)$ — ⓑ ① method

$\delta(q_0, a, A) = (q_1, \epsilon)$ — ⓒ ② method

Sol:- $q_k = \{q_0, q_1\}$ $q_l = \{q_0, q_1\}$

ⓐ $(q_0 Z q_0) \xrightarrow{q_k = q_0} a(q_0 A q_0)\ (q_0 Z q_0) \quad q_L = q_1,\ q_k \to 0$

$a(q_0 A q_1)\ (q_1 Z q_0) \quad q_L = q_1,\ q_k$

$(q_0 Z q_1) \xrightarrow{q_k = q_1} a(q_0 A q_0)\ (q_0 Z q_1) \quad q_L = q_0,\ q_k = 1$

$a(q_0 A q_1)\ (q_1 Z q_1)$

$q_L = q_1$

$S \rightarrow AS/BC$

(b) $(q_0 A q_0) \xrightarrow{q_k = q_0} b(q_0 A q_0)(q_0 A q_0) \,/$
$b(q_0 A q_1)(q_1 A q_0)$

Mangal Date / /20

$(q_0 A q_1) \xrightarrow{q_k = q_1} b(q_0 A q_0)(q_0 A q_1) \,/$
$b(q_0 A q_1)(q_1 A q_1)$

(c) $(q_0 A q_1) \rightarrow a$

(6) obtain CFG ~~which~~ that generates a language accepted by PDA

$\delta(q_0, a, z) = (q_0, Az)$ – (a) (1)
$\delta(q_0, b, A) = (q_1, \epsilon)$ – (b) (2)
$\delta(q_1, \epsilon, z) = (q_1, \epsilon)$ – (c) (2)
$\delta(q_0, a, A) = (q_3, \epsilon)$ – (d) (2)
$\delta(q_3, \epsilon, z) = (q_0, Az)$ (e) (1)

Soln.

$q_p = \{q_0, q_1, q_2, q_3\}$
$q_k = \{q_0, q_1, q_2, q_3\}$

(a) $(q_0 z q_0) \Rightarrow a(q_0 A q_0)(q_0 z q_0) / a(q_0 A q_1)(q_1 z q_0) /$
$a(q_0 A q_2)(q_2 z q_0) / a(q_0 A q_3)(q_3 z q_0)$

$(q_0 z q_1) \rightarrow a(q_0 A q_0)(q_0 z q_1) / a(q_0 A q_1)(q_1 z q_1) /$
$a(q_0 A q_2)(q_2 z q_1) / a(q_0 A q_3)(q_3 z q_1)$

$(q_0 z q_2) \rightarrow a(q_0 A q_0)(q_0 z q_2) / a(q_0 A q_1)(q_1 z q_2) /$
$a(q_0 A q_2)(q_2 z q_2) / a(q_0 A q_3)(q_3 z q_2)$

$(q_0 z q_3) \rightarrow a(q_0 A q_0)(q_0 z q_3) / a(q_0 A q_1)(q_1 z q_3) /$
$a(q_0 A q_2)(q_2 z q_3) / a(q_0 A q_3)(q_3 z q_3)$

Date / /20

(b) $(q_0 A q_1) \rightarrow b$

(c) $(q_1 Z q_1) \rightarrow \epsilon$

(d) $(q_0 A q_3) \rightarrow a$

(e) $(q_3 Z q_0) \rightarrow$

- Building Pda for a given regular expression

- NPDA characteristics :

  - if (q, ϵ, A) is defined
    (q, a, A) should not be defined.

1) $\delta(q, a, X)$ must have only one definition

- CFG in GNF to PDA

  - Stack → input Output (state q_1 to q_1)

  compulsory $\delta(q_0, \epsilon, Z_0) \rightarrow (q_1, S Z_0)$

- PDA to CFG
  - $(q_i, a, Z) \rightarrow (q_j, AB)$

    $\Rightarrow (q_i Z q_K) \rightarrow a (q_j A q_l) (q_l B q_K)$

    l & K all states in Q

  - $(q_i, a, Z) \rightarrow (q_j, \epsilon)$

    $\Rightarrow (q_i Z q_j) \rightarrow a$.

Closure Properties of CFL

Many operations on Context Free Languages (CFL) guarantee to produce CFL. A few do not produce CFL. *Closure properties* consider operations on CFL that are guaranteed to produce a CFL. The CFL's are closed under *substitution*, *union*, *concatenation*, *closure (star)*, *reversal*, *homomorphism* and *inverse homomorphism*. CFL's are not closed under *intersection* (but the intersection of a CFL and a regular language is always a CFL), *complementation*, and *set-difference*.

I. Substitution:

By substitution operation, each symbol in the strings of one language is replaced by an entire CFL language.

Example:

$S(0) = \{a^nb^n \mid n \geq 1\}$, $S(1)=\{aa,bb\}$ is a substitution on alphabet $\Sigma = \{0, 1\}$.

Theorem:

If a substitution s assigns a CFL to every symbol in the alphabet of a CFL L, then s(L) is a CFL.

Proof:

Let $G = (V, \Sigma, P, S)$ be grammar for the CFL L. Let $G_a = (V_a, T_a, P_a, S_a)$ be the grammar corresponding to each terminal $a \in \Sigma$ and $V \cap V_a = \phi$. Then $G' = (V', T', P', S)$ is a grammar for s(L) where

- $V' = V \cup V_a$
- T' = union of T_a's all for $a \in \Sigma$
- P' consists of
  - All productions in any P_a for $a \in \Sigma$
  - The productions of P, with each terminal a is replaced by S_a everywhere a occurs.

Example:

$L = \{0^n1^n \mid n \geq 1\}$, generated by the grammar $S \to 0S1 \mid 01$, $s(0) = \{a^nb^m \mid m \leq n\}$, generated by the grammar $S \to aSb \mid A$; $A \to aA \mid ab$, $s(1) = \{ab, abc\}$, generated by the grammar $S \to abA$, $A \to c \mid \varepsilon$. Rename second and third S's to S_0 and S_1, respectively. Rename second A to B. Resulting grammars are:

$S \rightarrow 0S1 \mid 01$
$S_0 \rightarrow aS_0b \mid A;\ A \rightarrow aA \mid ab$
$S_1 \rightarrow abB;\ B \rightarrow c \mid \varepsilon$

In the first grammar replace 0 by S_0 and 1 by S_1. The resulted grammar after substitution is:
$S \rightarrow S_0SS_1 \mid S_0S_1 \qquad S_0 \rightarrow aS_0b \mid A;\ A \rightarrow aA \mid ab \qquad S_1 \rightarrow abB;\ B \rightarrow c \mid \varepsilon$

II. Application of substitution:

a. *Closure under union of CFL's L_1 and L_2:*

Use L={a, b}, s(a)=L_1 and s(b)=L_2. Then s(L)= $L_1 \cup L_2$.

How to get grammar for $L_1 \cup L_2$?

Add new start symbol S and rules $S \rightarrow S_1 \mid S_2$

The grammar for $L_1 \cup L_2$ is G = (V, T, P, S) where V = $\{V_1 \cup V_2 \cup S\}$, $S \notin (V_1 \cup V_2)$ and P = $\{P_1 \cup P_2 \cup \{S \rightarrow S_1 \mid S_2\}\}$

Example:

$L_1 = \{a^nb^n \mid n \geq 0\}$, $L_2 = \{b^na^n \mid n \geq 0\}$. Their corresponding grammars are

G_1: $S_1 \rightarrow aS_1b \mid \varepsilon$, G_2 : $S_2 \rightarrow bS_2a \mid \varepsilon$

The grammar for $L_1 \cup L_2$ is

$G = (\{S, S_1, S_2\}, \{a, b\}, \{S \rightarrow S_1 \mid S_2, S_1 \rightarrow aS_1b \mid \varepsilon, S_2 \rightarrow bS_2a\}, S)$.

b. *Closure under concatenation of CFL's L_1 and L_2:*

Let L={ab}, s(a)=L_1 and s(b)=L_2. Then s(L)=L_1L_2

How to get grammar for L_1L_2?

Add new start symbol and rule $S \rightarrow S_1S_2$

The grammar for L_1L_2 is G = (V, T, P, S) where V = $V_1 \cup V_2 \cup \{S\}$, $S \notin V_1 \cup V_2$ and P = $P_1 \cup P_2 \cup \{S \rightarrow S_1S_2\}$

Example:

$L_1 = \{a^n b^n \mid n \geq 0\}$, $L_2 = \{b^n a^n \mid n \geq 0\}$ then $L_1L_2 = \{a^n b^{\{n+m\}} a^m \mid n, m \geq 0\}$

Their corresponding grammars are

$G_1: S_1 \rightarrow aS_1b \mid \varepsilon$, $G_2 : S_2 \rightarrow bS_2a \mid \varepsilon$

The grammar for L_1L_2 is

$G = (\{S, S_1, S_2\}, \{a, b\}, \{S \rightarrow S_1S_2, S_1 \rightarrow aS_1b \mid \varepsilon, S_2 \rightarrow bS_2a\}, S)$.

c. *Closure under Kleene's star (closure * and positive closure* $^+$*) of CFL's* L_1*:*

Let $L = \{a\}^*$ (or $L = \{a\}^+$) and $s(a) = L_1$. Then $s(L) = L_1^*$ (or $s(L) = L_1^+$).

Example:

$L_1 = \{a^n b^n \mid n \geq 0\}$ $(L_1)^* = \{a^{\{n1\}} b^{\{n1\}} \ldots a^{\{nk\}} b^{\{nk\}} \mid k \geq 0$ and $n_i \geq 0$ for all $i\}$

$L_2 = \{a^{\{n^2\}} \mid n \geq 1\}$, $(L_2)^* = a^*$

How to get grammar for $(L_1)^*$:

Add new start symbol S and rules $S \rightarrow SS_1 \mid \varepsilon$.

The grammar for $(L_1)^*$ is

$G = (V, T, P, S)$, where $V = V_1 \cup \{S\}$, $S \notin V_1$, $P = P_1 \cup \{S \rightarrow SS_1 \mid \varepsilon\}$

d. Closure under homomorphism of CFL L_i ***for every*** $a_i \in \Sigma$***:***

Suppose L is a CFL over alphabet Σ and h is a homomorphism on Σ. Let s be a substitution that replaces every $a \in \Sigma$, by h(a). ie $s(a) = \{h(a)\}$. Then $h(L) = s(L)$. ie $h(L) = \{h(a_1) \ldots h(a_k) \mid k \geq 0\}$ where $h(a_i)$ is a homomorphism for every $a_i \in \Sigma$.

III. Closure under Reversal:

L is a CFL, so L^R is a CFL. It is enough to reverse each production of a CFL for L, i.e., to substitute each production $A \rightarrow \alpha$ by $A \rightarrow \alpha^R$.

IV. Intersection:

The CFL's are not closed under intersection

Example:

The language $L = \{0^n 1^n 2^n \mid n \geq 1\}$ is not context-free. But $L_1 = \{0^n 1^n 2^i \mid n \geq 1, i \geq 1\}$ is a CFL and $L_2 = \{0^i 1^n 2^n \mid n \geq 1, i \geq 1\}$ is also a CFL. But $L = L1 \cap L_2$.

Corresponding grammars for L_1: $S \rightarrow AB$; $A \rightarrow 0A1 \mid 01$; $B \rightarrow 2B \mid 2$ and corresponding grammars for L_2: $S \rightarrow AB$; $A \rightarrow 0A \mid 0$; $B \rightarrow 1B2 \mid 12$.

However, $L = L_1 \cap L_2$, thus intersection of CFL's is not CFL

a. Intersection of CFL and Regular Language:

Theorem: If L is CFL and R is a regular language, then $L \cap R$ is a CFL.

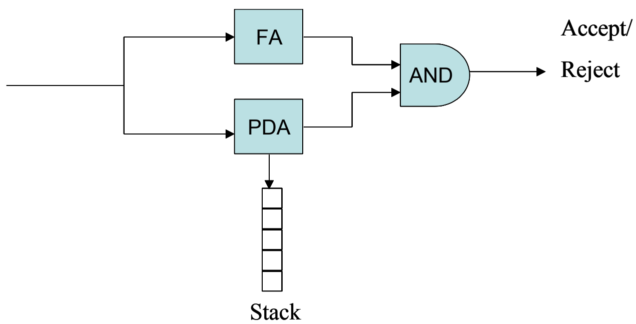


Figure 1: PDA for $L \cap R$

Proof:

$P = (Q_P, \Sigma, \Gamma, \delta_P, q_P, Z_0, F_P)$ be PDA to accept L by final state. Let $A = (Q_A, \Sigma, \delta_A, q_A, F_A)$ for DFA to accept the Regular Language R. To get $L \cap R$, we have to run a Finite Automata in parallel with a push down automata as shown in figure 1. Construct PDA $P' = (Q, \Sigma, \Gamma, \delta, q_o, Z_0, F)$ where

- $Q = (Q_p \text{ X } Q_A)$
- $q_o = (q_p, q_A)$
- $F = (F_P \text{X } F_A)$
- δ is in the form $\delta((q, p), a, X) = ((r, s), g)$ such that
  1. $s = \delta_A(p, a)$
  2. (r, g) is in $\delta_P(q, a, X)$

That is for each move of PDA P, we make the same move in PDA P′ and also we carry along the state of DFA A in a second component of P′. P′ accepts a string w if and only if both P and A accept w. ie w is in $L \cap R$. The moves $((q_p, q_A), w, Z) \mid\!-^{*}_{P'} ((q, p), \varepsilon, \gamma)$ are possible if and only if $(q_p, w, Z) \mid\!-^{*}_{P} (q, \varepsilon, \gamma)$ moves and $p = \delta^*(q_A, w)$ transitions are possible.

b. CFL and RL properties:

Theorem: The following are true about CFL's L, L_1, and L_2, and a regular language R.

1. **Closure of CFL's under set-difference with a regular language**. ie L - R is a CFL.

   Proof:

   R is regular and regular language is closed under complement. So R^C is also regular. We know that $L - R = L \cap R^C$. We have already proved the closure of intersection of a CFL and a regular language. So CFL is closed under set difference with a Regular language.

2. **CFL is not closed under complementation**

   L^C is not necessarily a CFL

   Proof:

   Assume that CFLs were closed under complement. ie if L is a CFL then L^C is a CFL. Since CFLs are closed under union, $L_1^C \cup L_2^C$ is a CFL. By our assumption $(L_1^C \cup L_2^C)^C$ is a CFL. But $(L_1^C \cup L_2^C)^C = L_1 \cap L_2$, which we just showed isn't necessarily a CFL. Contradiction! . So our assumption is false. CFL is not closed under complementation.

3. **CFLs are not closed under set-difference**. ie $L_1 - L_2$ is not necessarily a CFL.

   Proof:

   Let $L1 = \Sigma^* - L$. Σ^* is regular and is also CFL. But $\Sigma^* - L = L^C$. If CFLs were closed under set difference, then $\Sigma^* - L = L^C$ would always be a CFL. But CFL's are not closed under complementation. So CFLs are not closed under set-difference.

V. Inverse Homomorphism:

Recall that if h is a homomorphism, and L is any language, then $h^{-1}(L)$, called an *inverse homomorphism,* is the set of all strings w such that $h(w) \in L$. The CFL's are closed under inverse homomorphism.

Theorem:

If L is a CFL and h is a homomorphism, then $h^{-1}(L)$ is a CFL

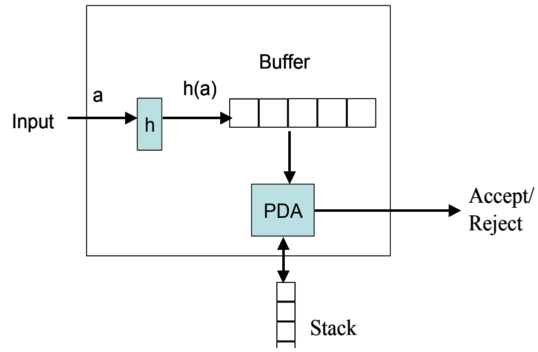


Figure 2: PDA to simulate inverse homomorphism

We can prove closure of CFL under inverse homomorphism by designing a new PDA as shown in figure 2. After input a is read, h(a) is placed in a buffer. Symbols of h(a) are used one at a time and fed to PDA being simulated. Only when the buffer is empty does the PDA read another of its input symbol and apply homomorphism to it.

Suppose h applies to symbols of alphabet Σ and produces strings in T*. Let PDA P = (Q, T, Γ, δ, q_0, Z_0, F) that accept CFL L by final state. We construct a new PDA P′ = (Q′, Σ, Γ, δ′, (q_0, ε), Z_0, (F x ε)) to accept $h^{-1}(L)$, where

- Q′ is the set of pairs (q, x) such that
  - q is a state in Q
  - x is a suffix of some string h(a) for some input string a in Σ
- δ′ is defined by
  - δ′ ((q, ε), a, X) = {((q, h(a)),X)}
  - If δ(q, b, X) = {(p, γ)} where b∈ T or b = ε then δ′ ((q, bx), ε, X) = {((p, x), γ)}
- The start state of P′ is (q_0, ε)
- The accepting state of P′ is (q, ε), where q is an accepting state of P.

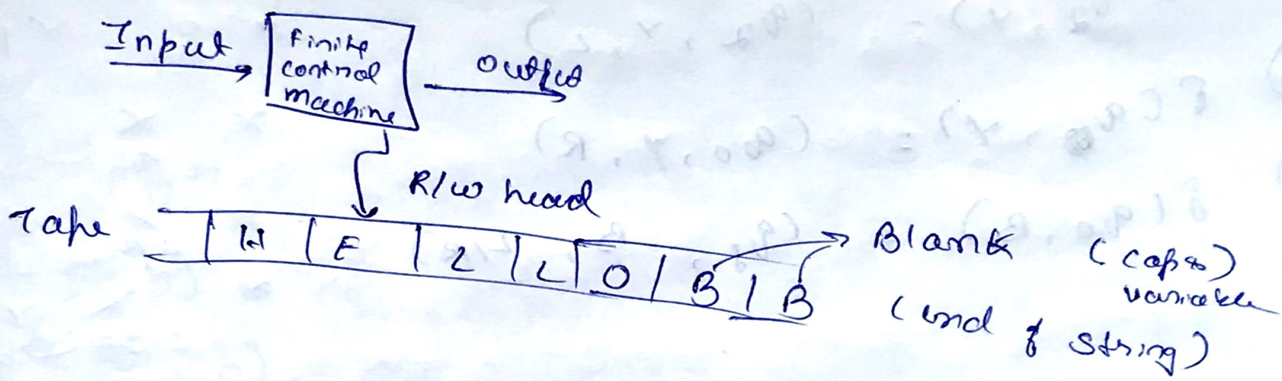
Once we accept the relationship between P and P′, P accepts h(w) if and only if P′ accepts w, because of the way the accepting states of P′ are defined.

Thus $L(P') = h^{-1}(L(P))$

UNIT – V

Mangal
Date / /20

Turing machine



$(Q, \Sigma, \Gamma, \delta, q_0, B, F)$

Γ = Tape symbols

B = Blank special tape character

Transition function

1) $\delta(q_0, a) = (q_1, a, R)$ → no change in input symbol a, go Right
2) $\delta(q_0, a) = (q_2, a, L)$ → no change in input symbol & go left
3) $\delta(q_0, a) = (q_1, x, R)$ → input symbol replaced by x & go Right
4) $\delta(q_0, a) = (q_1, x, L)$ → — " — left
5) $\delta(q_0, a) = (q_0, a, R)$ → No change in state & input symbol a, go right

Q) $L = \{0^n 1^n\}$ $n \geq 0$

1) $\delta(q_0, 0) = (q_1, X, R)$
2) $\delta(q_1, 0) = (q_1, 0, R)$
3) $\delta(q_1, 1) = (q_2, Y, L)$
4) $\delta(q_2, 0) = (q_2, 0, L)$

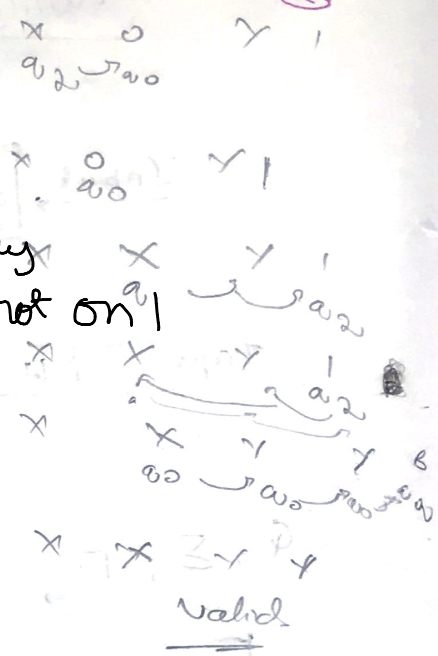
5) $\delta(q_2, x) = (q_0, x, R)$

6) $\delta(q_1, Y) = (q_1, Y, R)$

7) $\delta(q_2, Y) = (q_2, Y, L)$

8) $\delta(q_0, Y) = (q_0, Y, R)$ skip only on Y not on 1

9) $\delta(q_0, B) = (q_f, B, L/R)$



Q) $L = \{0^n 1^n 2^n\}$

$\delta(q_0, 0) \rightarrow (q_1, X, R)$

$\delta(q_1, 0) \rightarrow (q_1, 0, R)$

$\delta(q_1, Y) \rightarrow (q_1, Y, R)$

$\delta(q_1, 1) \rightarrow (q_2, Y, R)$

$\delta(q_2, 1) \rightarrow (q_2, 1, R)$

$\delta(q_2, Z) \rightarrow (q_2, Z, R)$

$\delta(q_2, 2) \rightarrow (q_3, Z, L)$

$\delta(q_3, 1) \rightarrow (q_3, 1, L)$

$\delta(q_3, Y) \rightarrow (q_3, Y, L)$

$\delta(q_3, 0) \rightarrow (q_3, 0, L)$

$\delta(q_3, X) \rightarrow (q_0, X, R)$

skip only for Y and Z not for 1 & 2

$\delta(q_0, Y) \rightarrow (q_0, Y, R)$

$\delta(q_0, Z) \rightarrow (q_0, Z, R)$

$\delta(q_0, B) \rightarrow (q_6, B, R)$

left side zeros are replaced with Blank
right side zeros are replaced with 1's

Mangal Date / /20

Monus operation

$$m \dot{-} n = \begin{cases} m-n, & \text{if } m > n \\ 0, & \text{if } m \le n \end{cases}$$

Q) $0^m 1 0^n$ $m > n$

1) $\delta(q_0, 0) = (q_1, B, R)$
2) $\delta(q_1, 0) = (q_1, 0, R)$
3) $\delta(q_1, 1) = (q_2, 1, R)$ → delimiter
4) $\delta(q_2, 0) = (q_3, 1, L)$
5) $\delta(q_3, 1) = (q_3, 1, L)$
6) $\delta(q_3, 0) = (q_3, 0, L)$
7) $\delta(q_3, B) = (q_0, B, R)$
8) $\delta(q_2, 1) = (q_2, 1, R)$ → replaced 1's
9) $\delta(q_2, B) = (q_4, B, L)$
10) $\delta(q_4, 1) = (q_4, B, L)$
11) $\delta(q_4, 0) = (q_4, 0, L)$
12) $\delta(q_4, B) = (q_0, 0, L)$
13) $\delta(q_0, 1) = (q_5, B, R)$ → delimiter
14) $\delta(q_5, 1) = (q_5, B, R)$

~~Q) m<n~~
~~Same till (2)~~
~~(3)~~

15) $\delta(q_5, 0) = (q_5, B, R)$
16) $\delta(q_5, B) = (q_f, B, R)$ α

~~001 0000~~

m-n BB BBBBB

00 00 1 00 B
q_1

B 000 1 00 B
q_1

B 000 1 00 B
q_2

B 000 1 1 0 B
q_3

B 000 1 1 0 B
q_0

B B 00 1 1 0 B
q_0

B B 00 1 1 0 B
q_2

B B 00 1 1 1 B

B B 00 1 1 1 B
q_3

B B 0 0 1 1 1 B
q_0

B B B 0 1 1 1 B

B B B 0 1 1 1 B
q_2

B B B 0 1 1 1 B
q_2

B B B 0 1 1 1 B
q_4

B B B 0 B B B B
q_4

B B B 0 B B B B
q_4

B B B 0 B B B B
q_4

B B B 0 B B B B
q_4

Mangal Date / /20

W = 00010

$q_0 00010 B \vdash B q_1 0010 B \vdash B 0 q_1 010 B \vdash$

$B 00 q_1 1 0 B \vdash B 001 q_2 0 B \vdash B 00 1 q_3 1 1 B$

$B 0 q_3 0 1 1 B \vdash B q_3 0 0 1 1 B \vdash q_3 B 0011 B \vdash$

$B q_0 0011 B \vdash$ ~~BBq₀011B~~ $B B q_1 0 1 1 B \vdash$

$B B 0 q_1 1 1 B \vdash B B 0 1 q_2 1 B \vdash B B 0 1 1 q_2 B \vdash$

$B B 0 1 q_4 1 B \vdash B B 0 q_4 1 B B \vdash B B q_4 0 B B \vdash$

$B q_4 B 0 B B B \vdash B 0 q_f 0 B B B$

Q) Design turing machine which accepts Palindrome

1) $\delta(q_0, 0) = (q_1, X, R)$
2) $\delta(q_1, 1) = (q_1, 1, R)$
3) $\delta(q_1, 0) = (q_1, 0, R)$
4) $\delta(q_1, B) = (q_3, B, L)$
5) $\delta(q_3, 0) = (q_4, X, L)$
6) $\delta(q_4, 0) = (q_4, 0, L)$ → $\delta(q_4, Y) = (q_0, Y, L)$
7) $\delta(q_4, 1) = (q_4, 1, L)$
8) $\delta(q_4, X) = (q_0, X, R)$
9) $\delta(q_0, 1) = (q_2, Y, R)$
10) $\delta(q_2, 1) = (q_2, 1, R)$
11) $\delta(q_2, 0) = (q_2, 0, R)$
12) $\delta(q_2, X) = (q_5, X, L)$
13) $\delta(q_2, Y) = (q_5, Y, L)$

14) $\delta(q_2, B) = (q_5, B, L)$

15) $\delta(q_5, 1) = (q_4, 1, L)$

16) $\delta(q_5, 0) = (q_4, 0, L)$

Q) Designing turing machine which perform multiplication operation

1) $\delta(q_0, 0) = (q_1, B, R)$
2) $\delta(q_1, 0) = (q_1, 0, R)$
3) $\delta(q_1, 1) = (q_2, 1, R)$
4) $\delta(q_2, 0) = (q_3, X, R)$
5) $\delta(q_3, 0) = (q_3, 0, R)$
6) $\delta(q_3, 1) = (q_4, 1, R)$
7) $\delta(q_4, B) = (q_5, 0, L)$
8) $\delta(q_5, 0) = (q_5, 0, L)$
9) $\delta(q_5, 1) = (q_5, 1, L)$
10) $\delta(q_5, X) = (q_2, X, R)$
11) $\delta(q_4, 0) = (q_4, 0, R)$
12) $\delta(q_2, 1) = (q_6, 1, L)$
13) $\delta(q_6, X) = (q_6, 0, L)$
14) $\delta(q_6, 1) = (q_6, 1, L)$
15) $\delta(q_6, 0) = (q_6, 0, L)$
16) $\delta(q_6, B) = (q_0, B, R)$
17) ~~$\delta(q_0, 1) = (q_7, B, R)$~~
18) $\delta(q_7, 0) = (q_7, B, R)$
19) $\delta(q_7, 1) = (q_7, B, R)$

B001001BB BB..
q_1

B001001BB BB
q_2

B001X01BB BB
q_3

B001X01BBBB
q_3

B001X01BBBB
q_4

B001X01 0BBB
q_5

B001X010BBB
q_5

B001X010BBB
q_2

B001XX10BBB
q_3

B001XX10BBB
q_4 (q_5)

B001XX100BB
q_5

B001XX100BB
q_2

B001XX100BB...
q_2

B001XX100BB
q_6

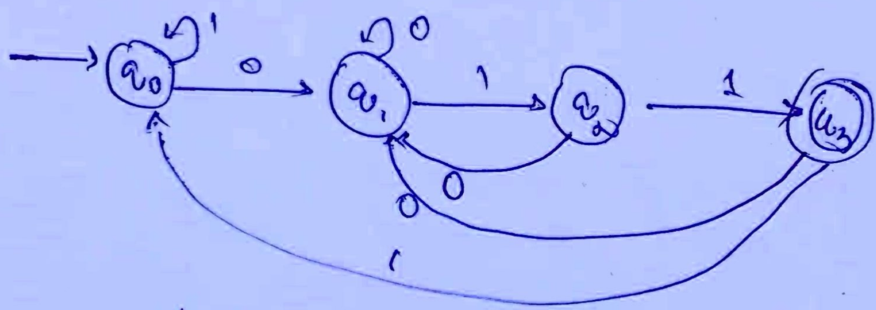
B001X0100BB
q_6

B 0 0 1 DD1 0 0 B B
$\uparrow$
q_6

B 0 0 1 00 1 0 0 B B
$\uparrow$
q_6

B 0 0 1 001 0 0 B B
$\uparrow$
q_0

Q) Design a turing machine which accepts the language ending with 011



	0	1
$\rightarrow q_0$	q_1	q_0
q_1	q_1	q_2
q_2	q_1	q_3
$* q_3$	q_1	q_0

TM TT

	0	1	B
q_0	$(q_1, 0, R)$	$(q_0, 1, R)$	
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	
q_2	$(q_1, 0, R)$	$(q_3, 1, R)$	
q_3	$(q_1, 0, R)$	$(q_0, 1, R)$	(q_f, B, R)

$w = 1011$

Date / /20

$q_0 1011B \vdash \quad 1q_0 011B \vdash \quad 10q_1 11B \vdash \quad 101q_2 1B \vdash$

$1011q_3 B \vdash \quad 1011B q_f.$