Name- Vivek Kumar Mohanta Regd. No - 2141013166
Section - CSE (B)

ASSIGNMENT - 1

1. In theoritical computer Science, how important are computing theory and complexity theory? Describe their differences and how they are related to one another.

Ans:- Complexity Theory:- Computer Problems come in different varieties, some are easy and some are hard.

Computability Theory:- Certain basic problems cannot be solved by computers. Eg:- Determining a mathematical equation is True or False.

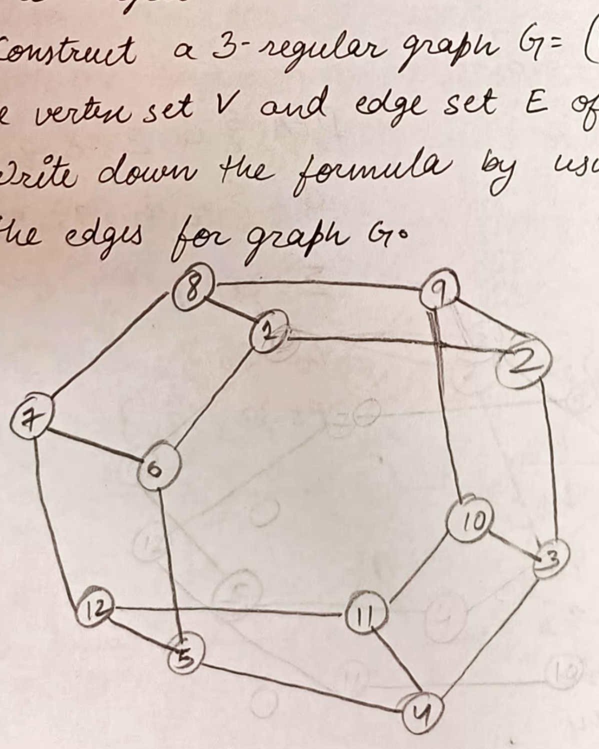
Complexity Theory & Computability Theory are closely related:- In complexity theory, the objective is to classify the problems as easy and hard ones, whereas in computability theory, the classification of problems is by those that are solvable and those that are not.

2- A graph G is said to be a K-regular if every node in the graph has a degree k.

a) Construct a 3-regular graph G= (V,E) with 12 nodes. Display the vertex set V and edge set E of the graph G.

b) Write down the formula by using which you constructed the edges for graph G.

Ans→ a)



Edge set E : { (1,2), (2,3), (3,4), (4,5), (5,6), (6,1),
(7,8), (8,9), (9,10), (10,11), (11,12), (12,7),
(6,7), (1,8), (2,9), (3,10), (4,11), (5,12) }

Vertex set V: { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12 }

b) Number of edges of a K-regular graph with N-vertices

$$= \frac{(N * K)}{2}$$

Here, N = 12, K = 3 ⇒ No. of edges = $\frac{12 \times 3}{2} = \frac{36}{2} = \underline{\underline{18}}$

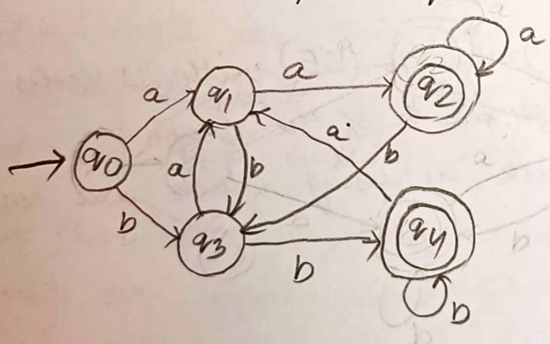
3. Construct the DFA for the following languages:

a) The language accepting all the strings such that the last two symbols must be same over input alphabets $\Sigma = \{a, b\}$

b) The language accepting all the strings that ends with 3 a's or 3 b's over input alphabets $\Sigma = \{a, b\}$

Ans:-

a)



DFA: { Q, Σ, δ, q_0, F }

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

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

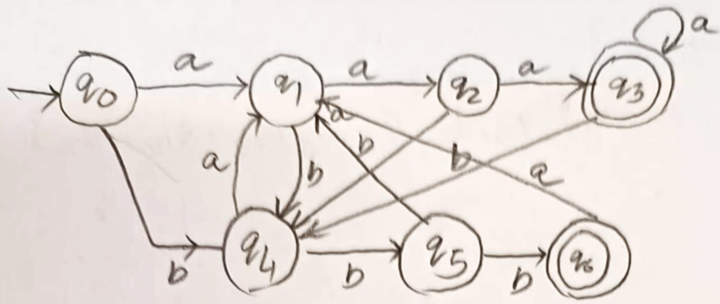
$q_0 = q_0$

$F = \{q_2, q_4\}$

Transition Mapping (δ):-

	a	b
q_0	q_1	q_3
q_1	q_2	q_3
q_2	q_2	q_3
q_3	q_1	q_4
q_4	q_1	q_4

b)



DFA = {Q, Σ, δ, q_0, F}

Q = { q_0, q_1, q_2, q_3, q_4, q_5, q_6 }

Σ = {a, b}

$q_0 = q_0$

F = { q_3, q_6 }

Transition Mapping (δ):-

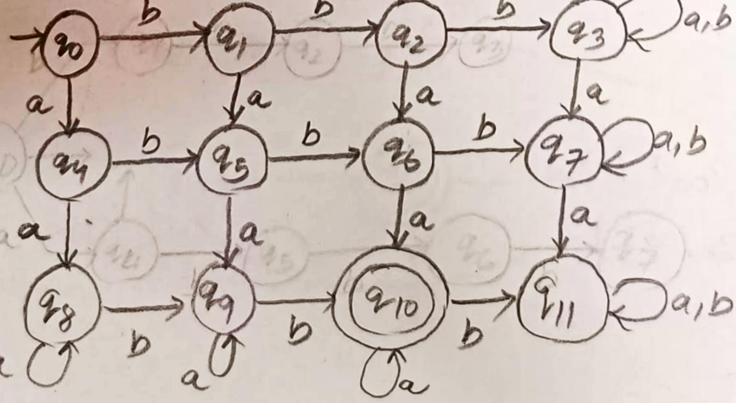
	a	b
q_0	q_1	q_4
q_1	q_2	q_4
q_2	q_3	q_4
q_3	q_3	q_4
q_4	q_1	q_5
q_5	q_1	q_6
q_6	q_1	q_6

4. Draw the state transition diagram and show the state transition table for the following DFA's:

a) DFA for the language accepting all strings that contains atleast 2 a's and exactly 2 b's over input alphabet Σ = {a, b}

b) DFA for the language strings containing neither '00' nor '11' as substring over input alphabets Σ = {0, 1}

Ans:- a)



$DFA = \{Q, \Sigma, \delta, q_0, F\}$

$Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8, q_9, q_{10}, q_{11}\}$

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

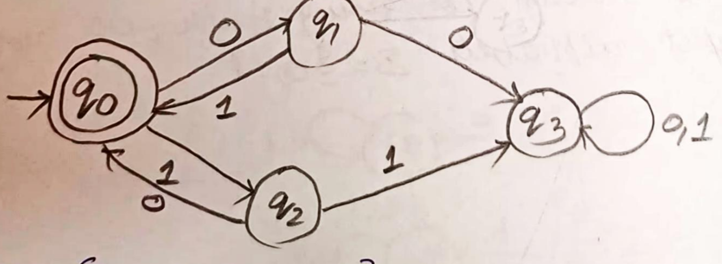
$q_0 = q_0$

$F = \{q_{10}\}$

Transition mapping : (δ) :-

	a	b
q_0	q_4	q_1
q_1	q_5	q_2
q_2	q_6	q_3
q_3	$\{q_3, q_7\}$	q_3
q_4	q_8	q_5
q_5	q_9	q_6
q_6	q_{10}	q_7
q_7	$\{q_7, q_{11}\}$	q_7
q_8	q_8	q_9
q_9	q_9	q_{10}
q_{10}	q_{10}	q_{11}
q_{11}	q_{11}	q_{11}

b)



$DFA = \{Q, \Sigma, \delta, q_0, F\}$

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

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

$q_0 = q_0$

$F = \{q_0\}$

Transition Mapping (δ):-

	0	1
q_0	q_1	q_2
q_1	q_3	q_0
q_2	q_0	q_3
q_3	q_3	q_3

5. a) Convert the following NFA with ε to NFA without ε.

b) Convert the obtained NFA to its equivalent DFA.

Ans:-

a)

$$q_0 \xrightarrow{\varepsilon^*} q_0 \xrightarrow{a} q_1 \xrightarrow{\varepsilon^*} q_1\, q_2$$

$$q_0 \xrightarrow{\varepsilon^*} q_0 \xrightarrow{b} \phi$$

$$q_1 \xrightarrow{\varepsilon^*} q_1 \xrightarrow{a} \phi$$

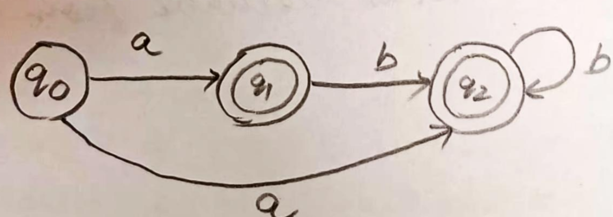
$$q_1 \xrightarrow{\varepsilon^*} q_1 \xrightarrow{b} q_2 \xrightarrow{\varepsilon^*} q_2$$

$$q_2 \xrightarrow{\varepsilon^*} q_2 \xrightarrow{a} \phi$$

$$q_2 \xrightarrow{\varepsilon^*} q_2 \xrightarrow{b} q_2 \xrightarrow{\varepsilon^*} q_2$$

	a	b
q_0	$\{q_1, q_2\}$	ϕ
q_1	ϕ	q_2
q_2	ϕ	q_2

$\Rightarrow$



b) Conversion from NFA to DFA

Transition mapping of NFA:-

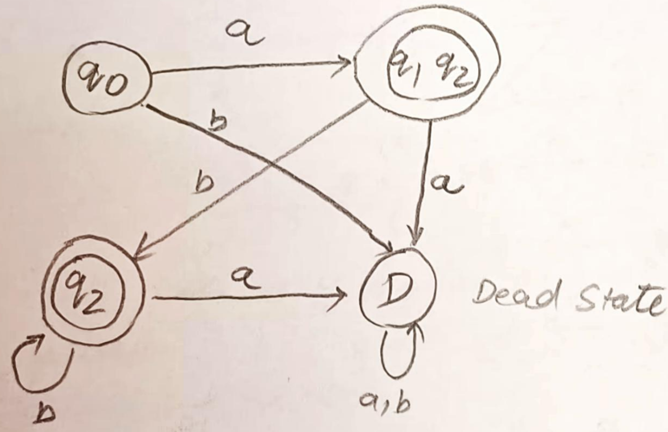
	a	b
q_0	$\{q_1, q_2\}$	ϕ
q_1	ϕ	q_2
q_2	ϕ	q_2

Transition mapping of DFA:-

	a	b
q_0	q_1q_2	ϕ
q_1q_2	ϕ	q_2
q_2	ϕ	q_2

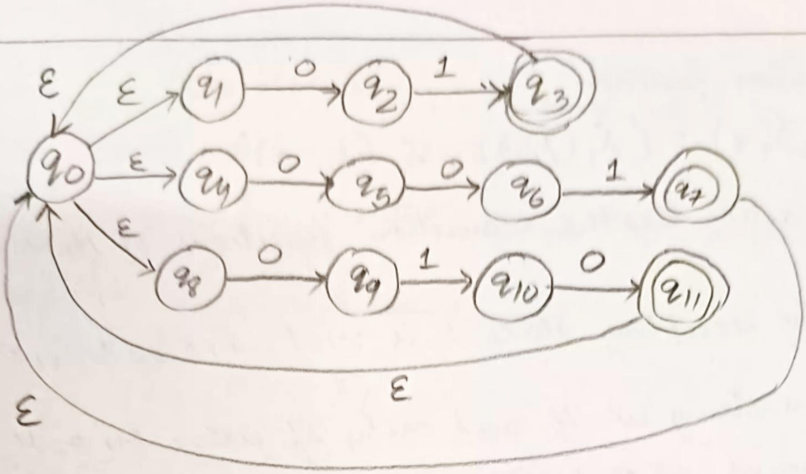
$\Rightarrow$ $q_0 = q_0$

$F = \{ q_1q_2 , q_2 \}$



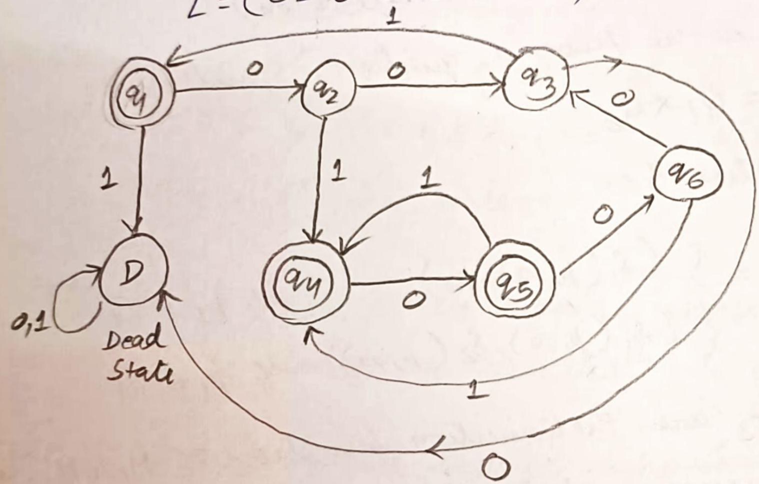
6- a) Design an NFA that recognizing the language $(01 \cup 001 \cup 010)^*$

b) Convert this NFA to its equivalent DFA. Give only the portion of the DFA that is reachable from the states.



b) NFA to DFA.

$L = (01 \cup 001 \cup 010)^*$



7. Prove that the class of Regular language is closed under:-

a) Union,

b) Concatenation, and

c) Kleene Closure.

Ans:- a) Let L_1 and L_2 be regular languages. This means that there exists deterministic finite automata (DFA) M_1 and M_2 that accept L_1 and L_2 respectively.

Let M be a DFA with states $Q = Q_1 \times Q_2$, where Q_1 and Q_2 are the set of states of M_1 and M_2 respectively. The start state of M is (q_{0_1}, q_{0_2}) where q_{0_1} is the start state of M_1 and q_{0_2} is the start state of M_2

The transition function δ is defined as:-

$$\delta((p_1, p_2), a) = (\delta_1(p_1, a), \delta_2(p_2, a))$$

where δ_1 and δ_2 are the transition functions of M_1 and M_2 respectively

The set of accepting state F is :- $F = F_1 \times Q_2 \cup Q_1 \times F_2$

$\therefore$ M accepts a string 'w' if and only if either M_1 or M_2 accepts 'w' which means $w \in L_1 \cup L_2$. Thus $L_1 \cup L_2$ is regular.

b) To Show $L_1 \cdot L_2$ is regular (concatenation)

As we have done in the previous question, similarly :-

$$Q = Q_1 \times Q_2$$

Transition function δ :-

$$\delta((p_1, p_2), a) = \begin{cases} (\delta_1(p_1, a), p_2) & \text{if } p_1 \notin F_1 \\ (\delta_1(p_1, a), \delta_2(p_2, a)) & \text{if } p_1 \in F_1 \end{cases}$$

where δ_1 and δ_2 are the transition function of M_1 & M_2 and F_1 is the set of accepting states of M_1.

$\therefore$ F is defined as :- $F = Q_1 \times F_2$ where F_2 is the set of accepting states of M_2.

Now, M accepts a string w if and only if M_1 accepts a prefix of 'w' and M_2 accepts the remaining part of 'w'.

$$\Rightarrow \quad w \in L_1 \cdot L_2, \text{ Thus } L_1 \cdot L_2 \text{ is regular.}$$

c) To show that Kleene closure L^* is regular, we can modify M to accept L^*

Let M' be a DFA with an additional state q_0' and a new ~~state~~ set of accepting state F'.

$F' = \{q_0'\} \cup F$ and F is the accepting states of M.

Transition function δ' :-

$$\delta'(q_0', \varepsilon) = q_0$$

where q_0 is the start state of M.

The rest of the transition remains the same as M.

Now, M' accepts a string 'w' if and only if w can be split into a concatenation of strings $w_1, w_2 \cdots w_k$, where each $w_i \in L$ for $1 \le i \le k$, which means $w \in L^*$.

Thus, L^* is regular.

8. a) Let $\Sigma = \{a, b\}$. Write regular expression to define language consisting of strings w such that, w of length even.

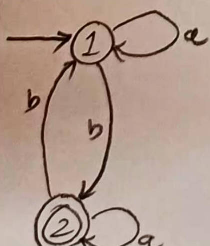
b) Let $\Sigma = \{a, b\}$. Write regular expression to define language consisting of strings w such that, w of length odd.

Ans:-

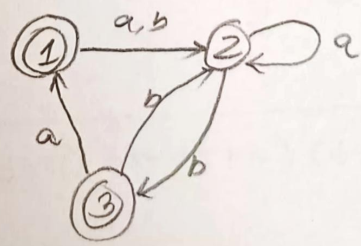
a) $R = [(a+b)(a+b)]^*$

b) $R = (a+b)[(a+b)(a+b)]^*$

9. Convert the following finite automata to regular expression:-



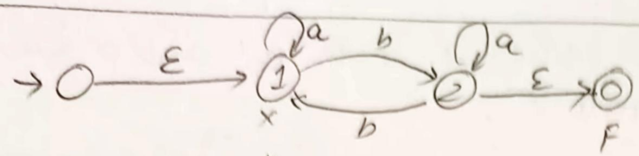
(a)



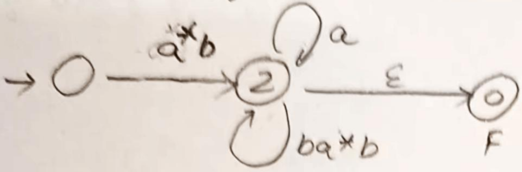
(b)

Ans:

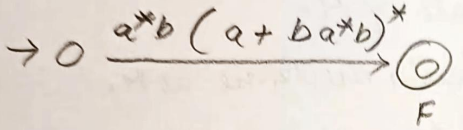
a)



⇓

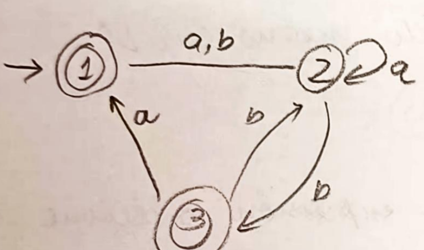


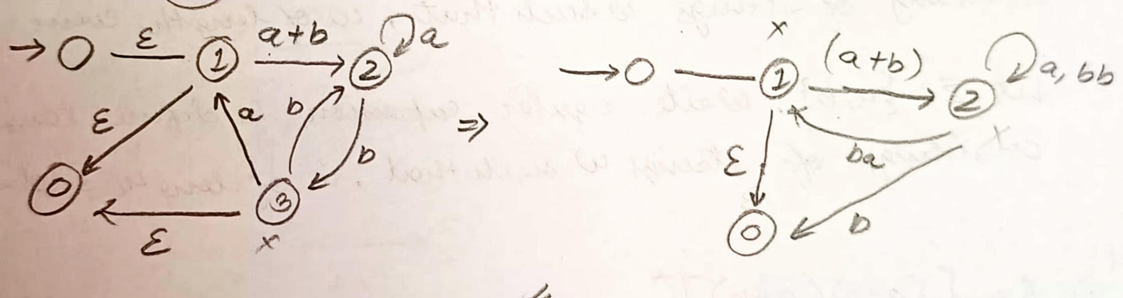
⇓

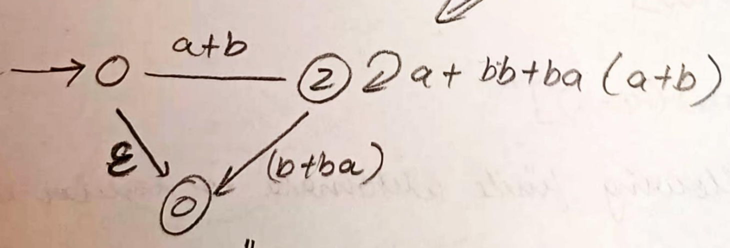


∴ The regular expression is :- $\boxed{R = a^*b\,(a+ba^*b)^*}$

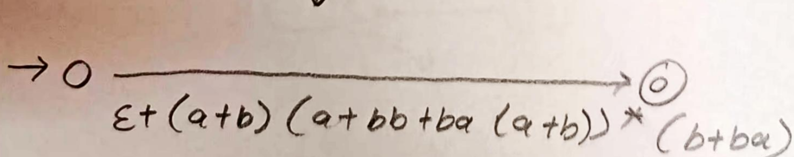
b)







⇓



∴ The regular expression is :- $R = \varepsilon+(a+b)(a+bb+ba(a+b))^*(b+ba)$

10- Design the Finite Automata for the following Regular Expression :-

i) $R1 = \phi$

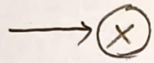
ii) $R_2 = \varepsilon$

iii) $R_3 = a^+$

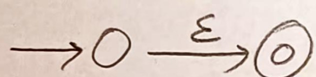
iv) $R_4 = (ab)^* ab^*$

v) $R5 = 0^* 1^*$

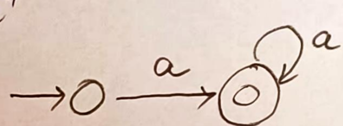
Ans:- i) $R_1 = \phi \Rightarrow L(R_1) = \{ \}$



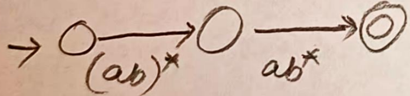
ii) $R_2 = \varepsilon \Rightarrow L(R_2) = \{ \}$



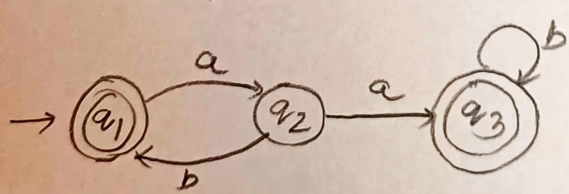
iii) $R_3 = a^+$



iv) $R_4 = (ab)^* ab^*$



⇓



v) $R5 = 0^* 1^*$