1. Explain the closure properties of R.E (10m)
2. List out applications of R.E (5m)
3. List out algebric clasue for R.E (Adulity rule R.E)
4. construct finite automata for a R.E (prblm)
   ↳ union, con, kleen
5. State & prove Arden's theorem
6. Construct RE for following FA (prblm).
7. write a short note on i) Regular lang ii) Regular Expre
8. Construct Regular lang for following R.E. $(a^+ b^+)$
9. Stmt of pumming lemma & Applications
10. check whether the following lang is regular lang or not based on pumming lemma
11. Find the equivalence of following 2 FA.
12. automata application

THEORY OF COMPUTATION

It's also known as automata theory

Auto → It self ; mata → machine or device

Automata is recognizing device / machine

- Automata is a recognizing device which is used to perform certain operation automatically without involving of humans

Example : i) Compiler → which act as translator HLL → MLL
  - check whole program
- Interpreter : check line by line code assembler
- convert HLL → MLL when we are having Add, sub

ii) Sensors : Eg: automatic door in mall

iii) Cellular communication Eg: Network, internet

iv) Robotics → which understand local human level language

*. **Alphabet** : An order list of symbols is said to be alphabet. It's denoted by Σ

Eg: $\Sigma_{english} = \{A, B, C \ldots Z\}$

$\Sigma_{Kannada} = \{$ಅ, ಆ, ... ಅಃ$\}$

$\Sigma_{Hindi} = \{$अ, आ, ... अः$\}$

* **String** : A group of symbols is said to be string. It's denoted by W.

Eg: W = 'kohli' , $W_1 = 0100$

*. **Length of the String**: Number of symbols is present in a string. It's denoted by $|W|$

$|W| = 5$ $|W_1| = 4$

* **Empty String** No symbols are present in string. It's denoted by ϵ (Epsilon)

Importance of Epsilon:

→ $x^0 = \epsilon$

let $x = ab$

$x = 2$

$2^0 = \epsilon$, $ab^0 = \epsilon$

→ $\epsilon \, \epsilon = \epsilon$

→ $\epsilon . NCET = NCET$

→ $\epsilon . \epsilon . \epsilon . \epsilon . \epsilon = \epsilon$

→ $\epsilon . a . \epsilon . b . \epsilon . c = abc$

→ ϵ. kohli . ϵ . rohit . ϵ = kohlirohit

* palindrome: If you read any string in forward or backward direction you will get same symbols

* Power of Alphabet

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

$\Sigma^0 = \{a, b\}^0 = \epsilon$

$\Sigma^1 = \{a, b\}^1 = \{a, b\}$

$\Sigma^2 = \{a, b\}^2 = \{aa, bb, ab, ba\}$

$\Sigma^3 = \{a, b\}^3 = \{aaa, aab, aba, bbb, baa, bab, bba, abb\}$

$\Sigma^n = \{n$ no of alphabet $\}$

$\Sigma^* = \{\epsilon^0 \cup \Sigma^1 \cup \Sigma^2 \cup \Sigma^3 \cup \Sigma^4 \ldots \Sigma^n\}$

$= \{\epsilon \cup \{a, b\} \cup (aa, ab, ba, bb) \ldots \Sigma^n\}$

$\Sigma^* = \{\epsilon, a, b, aa, ab, ba, bb \ldots\}$

$\Sigma^+ = \{\Sigma^1 \cup \Sigma^2 \cup \Sigma^3 \cup \Sigma^4 \cup \ldots \Sigma^n\}$

$= \{(a, b) \cup (aa, ab, ba, bb)\}$

$\Sigma^+ = \{a, b, aa, ab, ba, bb \ldots\}$

$\boxed{\Sigma^* = \epsilon + \Sigma^+}$

$\boxed{\Sigma^+ = \Sigma^* - \epsilon}$

→ Find power of Alphabet for following 2, 3

: $\Sigma = \{2, 3\}$

$\Sigma^0 = \{2, 3\}^0 = \epsilon$

$\Sigma^1 = \{2, 3\}^1 = \{2, 3\}$

$\Sigma^2 = \{2, 3\}^2 = \{22, 23, 33, 32\}$

$\Sigma^3 = \{2, 3\}^3 = \{222, 223, 232, 233, 333, 322, 323, 332\}$

$\Sigma^* = \{\epsilon \cup \Sigma^1 \cup \Sigma^2 \cup \Sigma^3\}$

$\Sigma^* = \{\epsilon, 2, 3, 22, 23, 33, 32, 222, 223, 232, 233, 333, 322, 323, 33$

$\Sigma^+ = \Sigma^* - \epsilon$

$\Sigma^+ = \{2, 3, 22, 23, 33, 32, 222, 223, 232, 233, 333, 322, 323, 332\}$

→ Language: It's a collection / group of string. It's denoted by 'L'

$L = \{a, b, ab, ba, bb \ldots\}$

Types of Languages:

1. Finite language: A group of countable strings
   eg: $\{a, b, aa\}$

2. Infinite: A group of uncountable string
   eg: $\{a, b, aa \ldots\}$

Operations of language:

1. Union
2. Intersection
3. Concatination
4. Compliment
5. Reverse
6. Positive closure
7. Kleen closure

1. Union: let 'L_1, L_2' be a 2 languages then their union notation is denoted by $L_1 \cup L_2$

Eg: $L_1 = \{0, 1, 00\}$; $L_2 = \{00, 11, 111\}$

$L_1 \cup L_2 = \{0, 1, 00, 00, 11, 111\}$.

2. Intersection: Let 'L_1, L_2' be a 2 languages then their intersection notation is denoted by $L_1 \cap L_2$

$L_1 \cap L_2 = \{\emptyset\}$ $\{00\}$

3. Concatination: Let '$L_1 L_2$' be a 2 languages then their intersection concatenation denoted by $L_1 L_2$

$L_1 = \{ab, ba, aa\}$ $L_2 = \{ba, ab, bb\}$

$L_1 L_2 = \{abba, baab, aabb, abab, abbb, baba, babb, aaba, aab\ldots$

Eg: $L_1 = \{ab, ba, bbb\}$ $L_2 = \{bab, abb\}$

$L_1 L_2 = \{abbab, ababb, babab, baabb, bbbbab, bbbabb\}$

4. Compliment: Let 'L' be a language then it's compliment notation is denoted by '$\bar{L}$' or 'L''

→ Find power of Alphabet for following two input symbols

$\Sigma = \{c, d, e\}$

$\Sigma^0 = \epsilon$, $\Sigma^1 = \{c, d, e\}$

$3^2=9$ $\Sigma^2 = \{cd, ce, dc, de, ec, ed, cc, dd, ee\}$

$3^3=27$ $\Sigma^3 = \{ccc, ddd, eee, cdd, cee, cde, ced, dcc,$
$dce, dee, dec, ecc, ecd, edc, eec, eed$
$ccd, cce,$

Language operation:

1. Find concatenation for following $x = ab$, $y = bb$, $z = ba$
then find i) xy, ii) x^2y
iii) xy^2 iv) xyz.

→ $xy = \{abbb\}$

$x^2y = (ab)^2 bb$ or $x.x.y$.
$= ababbb$

$xy^2 = ab(bb)^2$
$abbbbb$

$xyz = \{abbbba\}$.

Compliment:

Let L be a language then it's compliment notation is denoted by $\bar{L}$ or L'

Let L be a language then it's reverse notation is denoted by 'L^R'

$\bar{L}$ or $L' = L^* - L$
$= \{\epsilon, a, b, aa, bb, ...\} - \{aa\}$
$= \{\epsilon, a, b, bb, ...\}$

$L = \{ab, ba, baa, abb\}$
$L^R = \{ba, ab, aab, bba\}$

→ **Positive closure:** (Excluding ϵ)

Let 'L' be a language then it's notation is denoted by 'L^+'.

i.e $\boxed{L^+ = \bigcup_{i=1}^{n} L^i}$

length

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

$L^+ = \{L^1 \cup L^2 \cup L^3 \cup \dots\}$

$= \{(a/b)^1 \cup (a/b)^2 \cup (a/b)^3 \cup \dots\}$

$= \{a, b \cup aa, bb, ab, ba \dots\}$

$= \{a, b, aa, bb, ab, ba \dots\}$.

→ **Kleen closure:** (Including ϵ)

$\boxed{L^* = \bigcup_{i=0}^{n} L^i}$

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

$L^+ = \{L^0 \cup L^1 \cup L^2 \cup L^3 \cup \dots\}$

$= \{(a/b)^0 \cup (a/b)^1 \cup (a/b)^2 \cup \dots\}$

$= \{\epsilon, a, b, \cup, aa, bb, ab, ba \dots\}$

$= \{\epsilon, a, b, aa, bb, ab, ba\}$.

$\boxed{\begin{array}{l} L^+ = L^* - \epsilon \\ L^* = L^+ + \epsilon \end{array}}$

→ find concatenation for following

$x =$ abb, $y =$ baa, $z =$ aab

i) $x^3 y^2$

$(abb)^3 (baa)^2$

abbabbabbbaabaa

ii) $x^2 y^2 z$

$(abb)^2 (baa)^2$ aab

abbabbbaabaaaab

iii) $x^3 y z$

$(abb)^3$ baaaab

abbabbabbbaaaab

iv) $x y^3 z \Rightarrow abb(baa)^3 aab$

abbbaabaabaaaab

i) $ab(a/b)^0 \cup ab(a/b)^1 \cup ab(a/b)^2$ }

$= ab\epsilon \cup aba, abb, abaa$

abab, abba, abbb ... }

iii) $(a/b)^0 ba \cup (a/b)^1 ba \cup (a/b)^2 ba$

ba $\cup$, aba, abba, aaba,

abba, baba, bbba.

{abba, ababa, abbba,

abaaba, abaaba, abbaba,

abbbba ... }

→ Construct Language based on lang operation that accepts all strings of a's and b's in which Every string starts with 'a'

⇒ By using Kleen closure

$L^* = \bigcup_{i=0}^{\infty} L^i$

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

$L^* = \{L^0 \cup L^1 \cup L^2 \cup \ldots\}$

$= \{a\binom{a}{b}^0 \cup a\binom{a}{b}^1 \cup a\binom{a}{b}^2 \cup \ldots\}$

$= \{a(\epsilon) \cup a\binom{a}{b} \cup a(aa, ab, ba, bb\} \cup \ldots\}$

$= \{a \cup aa, ab \cup aaa, aab, aba, abb \ldots\}$

$= \{a, aa, ab, aaa, aab, aba, abb \ldots\}$

→ Construct language that accepts all strings of a's and b's starts with 0 and ends with 1

⇒ $L^* = \bigcup_{i=0}^{\infty} L^i$

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

$L^* = \{L^0 \cup L^1 \cup L^2 \cup \ldots\}$

~~$= \ldots \cup 0\binom{0}{1}^1 \cup 0(00, 01, 10, 11) \cup \ldots\}$~~

~~$= 0\epsilon\ 00, 01 \cup 000, 001, 010, 011) \cup \ldots\}$~~

$= \{0\binom{0}{1}^0 1 \cup 0\binom{0}{1}^1 1 \cup 0\binom{0}{1}^2 1 \cup \ldots\}$

$= \{0\epsilon 1 \cup 0\binom{0}{1}1 \cup 0(00, 01, 10, 00)1 \cup \ldots\}$

$= \{01 \cup (001, 011) \cup 0001, 0011, 0101, 0001) \cup \ldots\}$

$= \{01, 001, 011, 0001, 0011, 0101, 0001 \ldots\}$

→ Construct language that accepts all strings of a's & b's

i) in which every string starting with ab

ii) in which every string ended with ba

iii) start with ab & end with ba

⇒ $L^* = \{L^0 \cup L^1 \cup L^2 \cup \ldots\}$

iii) $= \{ab\binom{a}{b}^0 ba \cup ab\binom{a}{b}^1 ba \cup ab\binom{a}{b}^2 ba \cup \ldots\}$

$= ab(\epsilon)ba \cup ab\binom{a}{b}ba \cup ab(aa, ab, ba, bb)ba \cup \ldots\}$

$= abba \cup ababa, abbba \cup abaaba, ababba, abbaba, abbbba.\}$

→ Construct language that accepts all strings of a's and b's in which every string contain sub.string as ab

$L^* = \bigcup_{i=0}^{\infty} L^i \qquad \Sigma = \{a, b\}$

$L^* = \{L^0 \cup L^1 \cup L^2 - - - \}$

$L^* = \{ab\binom{a}{b}^0 \cup ab\binom{a}{b}^1 \cup ab\binom{a}{b}^2 - - - \}$

$= \{ab(\varepsilon) \cup ab(a, b) \cup ab(aa, bb, ab, ba)\}.$

$= \{(ab) \cup (aba, abb) \cup (abaa, abbb, abab, abba)\}$

$= \{ab, (aba, abb), (abaa, abbb, abab, abba)\}.$

$L^* = \bigcup_{i=0}^{\infty} L^i$

$= \{L^0(ab)L^0 \cup L^1(ab)L^1 \cup L^2(ab)L^2 \cup - - - \}$

$= \{\binom{a}{b}^0 ab \binom{a}{b}^0 \cup \binom{a}{b}^1 ab \binom{a}{b}^1 \cup \binom{a}{b}^2 ab \binom{a}{b}^2 - - \}$

$= \{\varepsilon ab \varepsilon \cup \binom{a}{b} ab \binom{a}{b} \cup \begin{pmatrix} ab \\ ba \\ bb \\ aa \end{pmatrix} ab \begin{pmatrix} aa \\ ab \\ ba \\ bb \end{pmatrix} - - - \}$

$= \{ab, (aaba, babb);$

→ Construct language that accepts all strings of 0's & 1's in which every string 3 consitituties of 0 & 3 constitutes of 1.

$L^* = \bigcup_{i=0}^{\infty} L^i$

$L^* = \{L^0 \cup L^1 \cup L^2 - - - \}$

$= \{000\binom{0}{1}^0 111 \cup 000\binom{0}{1}^1 111 \cup 000\binom{0}{1}^2 111 - - - \}$

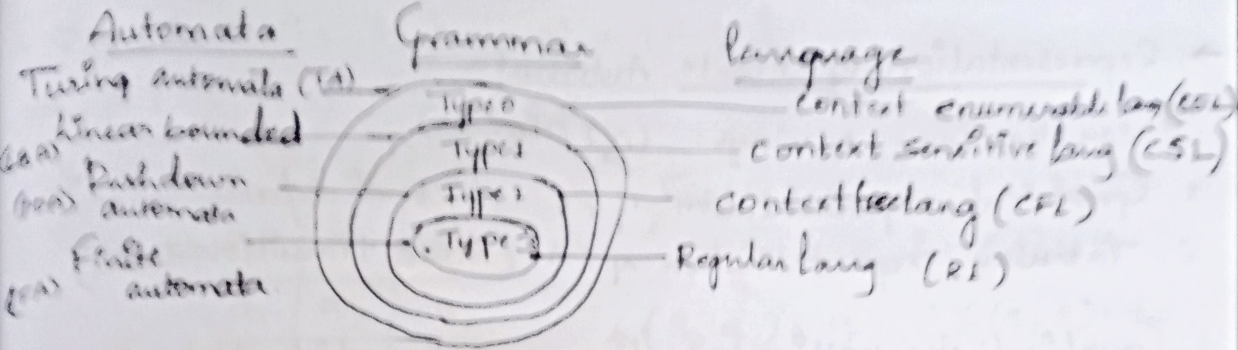
$= \{000\varepsilon 111 \cup 000\binom{0}{1}111 \cup 000\begin{pmatrix} 00 \\ 01 \\ 10 \\ 11 \end{pmatrix}111 - - \}$

$= \{000111;\ 0000111,\ 0001111,\ 00000111,\ 0000111$

$00010111,\ 00011111.\}.$

* Chamsky's Hiaarchy of languages:
(5m)

- chamsky is a name of the scientist
- Is mainly describe about the various types of lang's
- In automata theory.



- Type 3 Grammar lang is called 'Regular language'. Regular lang is accepted by 'Finite automata'.
- Type 2 Grammar lang is called 'Context free lang' which is accepted by 'Push down automata'
- Type 1 Grammar lang is called 'Context sensitive lang' which is accepted by 'Linear bounded automata'
- Type 0 Grammar lang is called 'Context enumerable' which is accepted by 'Turing automata'

(a) Finite state machine

→ Finite Automata: it's a mathematical representation of system.

- As per automata rule's it can be defined by

$M = \{Q, \Sigma, q_0, F, \delta\}$ — tuples

where, Q = No of states (countable)

Σ = Finite no of input alphabets / symbols

q_0 = Starting / Initial state

F = Final state

(delta) δ = Transition function

→ Types of Finite Automata: (FA or FSM)

1. Finite automata without output
   - Deterministic FA (DFA)
   - Non-deterministic (NFA)
   - ε-NFA
2. FA with output
   - Mealy machine
   - Moore machine

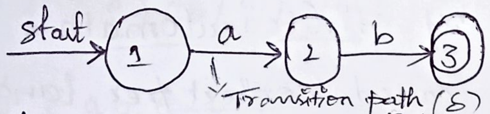
→ Representation of Finite Automata:

It can be done 2 ways: (S T Diagram)

1. Graphical representation of FA (Visual representation)
2. Tabular representation of FA: (State transition table)

1. **Graphical**: It's nothing but diagram in which states are represented by circle (O) & the transition are rep by →

Eg:



* final state is rep by double circle.

Formal definition of a FA

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

$Q = \{1, 2, 3\}$ no. of states

$\Sigma = \{a, b\}$ no of inputs

$q_0 = \{1\}$ starting state

$F = \{3\}$ final state

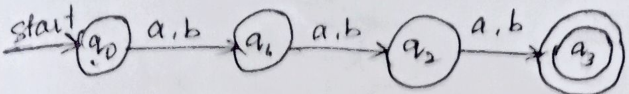
2. **Tabular**: It's nothing but rows and columns in which rows are rep by states (Q) and columns are rep input Symbols (Σ).

Eg:

Q \ Σ	a	b
→ 1	2	φ
2	φ	3
(3)*	φ	φ

* final state is rep by single circle or star
* Initial state is rep by single arrow.

1. Spec: Represent the graphical & Tabular representation for following diagram:



STD

→ Formal definition of FA

i.e. $M = \{Q, \Sigma, q_0, F, \delta\}$

$Q = \{q_0, q_1, q_2, q_3\}$; $\Sigma = \{a, b\}$

$q_0 = \{q_0\}$; $F = \{q_3\}$

STT

Q \ Σ	a	b
→ q_0	q_1	q_1
q_1	q_2	q_2
q_2	q_3	q_3
q_3*	ϕ	ϕ

$\delta = Q \times \Sigma$

$= \{q_0, q_1, q_2, q_3\} \times \{a, b\}$

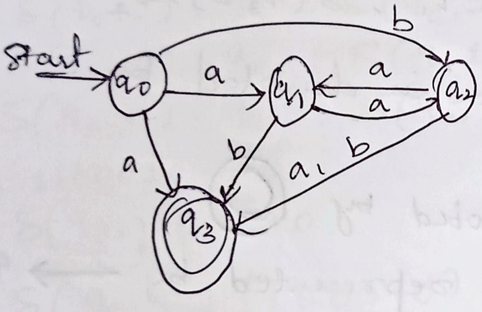
$\delta(q_0, a) = q_1$; $\delta(q_2, a) = q_3$

$\delta(q_0, b) = q_1$; $\delta(q_2, b) = q_3$

$\delta(q_1, a) = q_2$; $\delta(q_3, a) = \phi$

$\delta(q_1, b) = q_2$; $\delta(q_3, b) = \phi$

2.



Formal def of FA

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

$Q = \{q_0, q_1, q_2, q_3\}$; $\Sigma = \{a, b\}$

$q_0 = \{q_0\}$; $F = \{q_3\}$

STT:

Q \ Σ	a	b
→ q_0	q_1, q_3	q_2
q_1	q_2	q_3
q_2	q_1, q_3	q_3
q_3	ϕ	ϕ

$\delta = Q \times \Sigma$

$= \{q_0, q_1, q_2, q_3\} \times \{a, b\}$

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

$\delta(q_0, b) = q_2$

$\delta(q_1, a) = q_2$

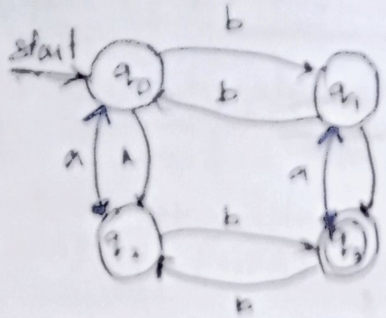
$\delta(q_1, b) = q_3$

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

$\delta(q_2, b) = q_3$

$\delta(q_3, a) = \phi$; $\delta(q_3, b) = \phi$

3.



Formal def of FA

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

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

$\Sigma = \{a, b\}$ $\quad F = \{q_3\}$

$Q \setminus \Sigma$	a	b
$\rightarrow q_0$	q_2	q_1
q_1	q_3	q_0
q_2	q_0	q_3
q_3	q_1	q_2

$\delta = Q \times \Sigma$

$\{q_0, q_1, q_2, q_3\} \times \{a, b\}$

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

$\delta(q_1, a) = q_3$; $\delta(q_1, b) = q_0$

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

$\delta(q_3, a) = q_1$; $\delta(q_3, b) = q_2$

NOTE: 1. In STD, the starting state is denoted by start $\rightarrow q_0$

2. In STD, final state is denoted by ◎

3. In STT, starting state is represented by $\rightarrow q_0$

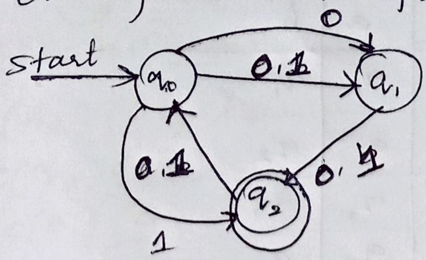
4. In STT, final state is rep by * q O

Acceptance of a String:

When we process language of all strings from starting state to till final state, if any string reached to the final state then we can say it as Acpt a string

Rejected String: If any string from starting state to the final state, if any string doesn't reached to the final state.

1. Check whether the following strings are Acceptance or rejected? for following final automata.



Strings:

i) 0110

ii) 011

iii) 0101

iv) 11101

v) 1100010

→ Formal def of FA

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

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

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

$q_0 = \{q_0\}$

$F = \{q_2\}$

STT

$Q \backslash \Sigma$	0	1
→ q_0	q_1	q_1, q_2
q_1	q_2	q_2
(q_2)	q_0	q_0

Transition ~~function~~:

$\delta = Q \times \Sigma \Rightarrow \{q_0, q_1, q_2\} \times \{0, 1\}$

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

$\delta(q_1, 0) = q_2$; $\delta(q_1, 1) = q_2$

$\delta(q_2, 0) = q_0$; $\delta(q_2, 1) = q_0$

v) 1100010

$\delta(q_0, 1100010)$

$\delta(q_1, q_2)(100010)$

$\delta(q_0, q_2)(00010)$

$\delta(q_0, q_1)(0010)$

$\delta(q_1, q_2)(010)$

$\delta(q_0, q_2)(10)$

$\delta(q_0, q_1, q_2)(0)$

$\delta(q_0, q_1, q_2) \in F$ → Accepted

i) 0110

$\delta(q_0, 0110)$

$\delta(q_1)(110)$

$\delta(q_2)(10)$

$\delta(q_0)(0)$

$\delta(q_1) \notin F$ ⇒ rejected

ii) 011

$\delta(q_0, 011)$

$\delta(q_1)(11)$

$\delta(q_2)(1)$

$\delta(q_0) \notin F$ ⇒ rejected

iii) 0101

$\delta(q_0, 0101)$

$\delta(q_1)(101)$

$\delta(q_2)(01)$

$\delta(q_0)(1)$

$\delta(q_1, q_2) \in F$ → Accepted

iv) 11101

$\delta(q_0, 11101)$

$\delta(q_1, q_2)(1101)$

$\delta(q_0, q_2)(101)$

$\delta(q_0, q_1, q_2)(01)$

$\delta(q_0, q_1, q_2)(1)$

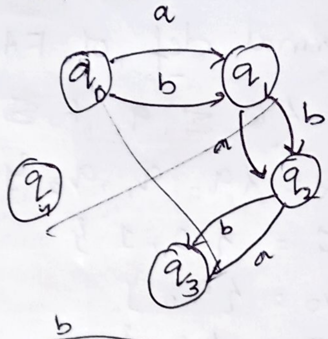
$\delta(q_0, q_1, q_2) \in F$ ⇒ Accepted

Q. Verify the following string whether appectance or rejected for following STT.

1) abab   iii) abbabab
ii) abbba   iv) aaaba
v) babababbaa

Q \ Σ	a	b
→ q_0	q_1	q_1
q_1	q_2	q_2
q_2*	q_3	q_3
q_3	q_0	q_1
q_4	q_2	q_3

→ Formal def of FA



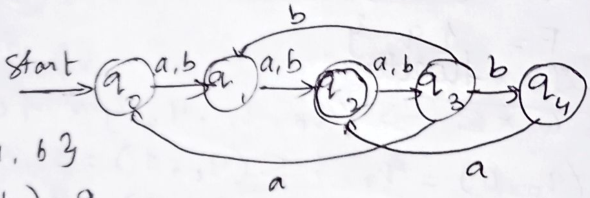
$M = \{Q, \Sigma, q_0, F, \delta\}$

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

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

$q_0 = \{q_0\}$

$F = \{q_2\}$



$\delta = Q \times \Sigma$

$= \{q_0, q_1, q_2, q_3, q_4\} \times \{a, b\}$

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

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

$\delta(q_2, a) = q_3$ $\delta(q_2, b) = q_3$

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

$\delta(q_4, a) = q_2$ $\delta(q_4, b) = q_3$

i) abab

$\delta(q_0, abab)$

$\delta(q_1)(bab)$

$\delta(q_2)(ab)$

$\delta(q_3)(b)$

$\delta(q_1) \Rightarrow \notin F$ = rejected

$q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{a} q_3 \xrightarrow{b} q_1 \notin F$

ii) abbba

$\delta(q_0, abbba)$

$\delta(q_1)(bbba)$

$\delta(q_2)(bba)$

$\delta(q_3)(ba)$

$\delta(q_1)(a)$

$\delta(q_2) \in F \Rightarrow$ accepted

$q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{b} q_3 \xrightarrow{b} q_1 \xrightarrow{a} q_2 \in$

iii) abbabab

$\delta(q_0)(abbabab)$

$\delta(q_1)(bbabab)$

$\delta(q_2)(babab)$

$\delta(q_3)(abab)$

$\delta(q_0)(bab)$

$\delta(q_1)(ab)$

$\delta(q_2)(b)$

$\delta(q_3) \notin F \Rightarrow$ rejected

$q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{b} q_3 \xrightarrow{a} q_0 \xrightarrow{b} q_1 \xrightarrow{a} q_2 \xrightarrow{b} q_3 \notin F$

iv) aaaba

$\delta(q_0, aaaba)$

$\delta(q_1)(aaba)$

$\delta(q_2)(aba)$

$\delta(q_3)(ba)$

$\delta(q_1)(a)$

$\delta(q_2) \in F \Rightarrow$ accepted

$q_0 \xrightarrow{a} q_1 \xrightarrow{a} q_2 \xrightarrow{a} q_3 \xrightarrow{b} q_1 \xrightarrow{a} q_2 \in F$

v) bababbaa

$\delta(q_0, bababbaa)$

$\delta(q_1)(ababbaa)$

$\delta(q_2)(babbaa)$

$\delta(q_3)(abbaa)$

$\delta(q_0)(bbaa)$

$\delta(q_1)(baa)$

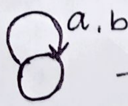
$\delta(q_2)(aa)$

$\delta(q_3)(a)$

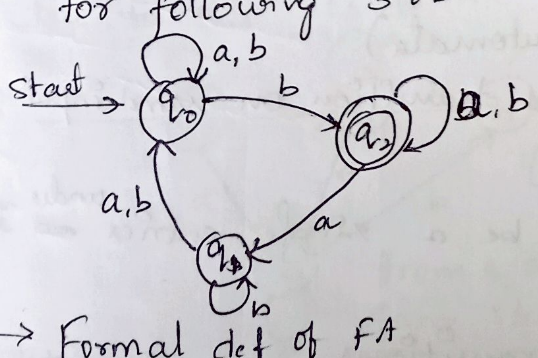
$\delta(q_0) \notin F =$ rejected

$q_0 \xrightarrow{b} q_1 \xrightarrow{a} q_2 \xrightarrow{b} q_3 \xrightarrow{a} q_0 \xrightarrow{b} q_1 \xrightarrow{b} q_2 \xrightarrow{a} q_3 \xrightarrow{a} q_0 \notin F$

→ Graphical Symbols of Automata:

1.  → State
2.  → Final state
3. Start → starting state
4.  → Either 1 a's or as (Self state of 'a'). many no of a's based on problem requirement.
5.  → Self State of 'b'.
6.  → Self state of 'a.b'.

1. check the following strings are accepted or rejected for following STD



→ Formal def of FA

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

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

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

$q_0 = \{q_0\}$

$F = \{q_2\}$

Q \ Σ	a	b
→ q_0	q_0	q_0, q_2
q_1	q_0	q_0, q_1
q_2	q_1, q_2	q_2

$\delta = Q \times \Sigma$

$= \{q_0, q_1, q_2\} \times \{a, b\}$

$\delta(q_0, a) = q_0$; $\delta(q_0, b) = q_0, q_2$

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

$\delta(q_2, a) = q_1 q_2$; $\delta(q_2, b) = q_2$.

i) ababab

$\delta(q_0, ababab)$

$\delta(q_0)(babab)$

$\delta(q_0, q_2)(abab)$

$\delta(q_0, q_1, q_2)(bab)$

$\delta(q_0, q_1, q_2)(ab)$

$\delta(q_0, q_1, q_2)(b)$

$\delta(q_0, q_1, q_2) \in F \Rightarrow$ accepted

ii) aaaba

$\delta(q_0, aaaba)$

$\delta(q_0)(aaba)$

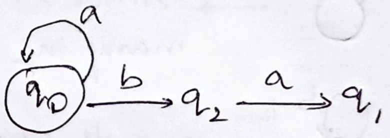
$\delta(q_0)(aba)$

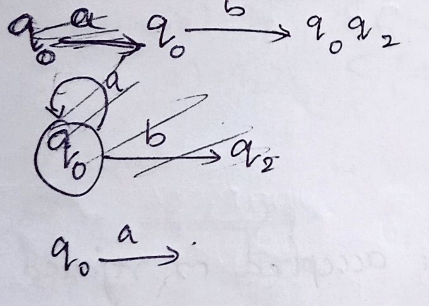
$\delta(q_0)(ba)$

$\delta(q_0, q_2)(a)$

$\delta(q_0, q_1, q_2) \in F$

= accepted





→ **DFA**: (deterministic finite automata)

- In DFA there must be only 1 transition on same input symbol.
- In STD of DFA there must be a single entries under the all input symbol.
- DFA doesn't allow ϵ transitions from state to another

Formal def of DFA:

- $M = \{Q, \Sigma, q_0, F, \delta\}$

$\boxed{\delta = Q \times \Sigma}$

- If there's no transition in case of DFA then we need to specify 'dead state'

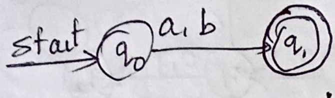
→ **NFA : (Non-deterministic finite automata)**

- In NFA, there may be more than 1 transitions from a state on same input symbol.
- In STD of NFA, there may be more than 1 entry under all input symbols.
- NFA, will allow ϵ transitions.

Formal def. of NFA:

$M = \{Q, \Sigma, q_0, F, \delta\}$.

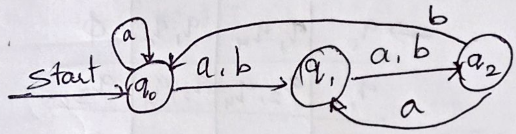
$$\boxed{\delta = 2^Q \times \Sigma}$$

1) 

(DFA)

STT

$Q \backslash \Sigma$	a	b
$\rightarrow q_0$	q_1	q_1
q_1^*	dead	dead state

2) 

(NFA) due to self loop on q_0 (a)

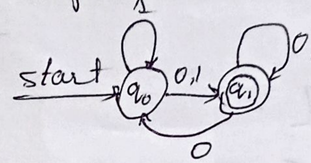
STT

$Q \backslash \Sigma$	a	b
$\rightarrow q_0$	q_0, q_1	q_1
q_1	q_2	q_2
q_2	q_1	q_0

$\rightarrow 2^Q = 2^a$

From q_0 state on same i/p symbol same state (q_0) & next state (q_1) due, to this it's NFA.

- If there's no transition from 1 to another state from any i/p symbol then it's denoted by 'ϕ'

3) 

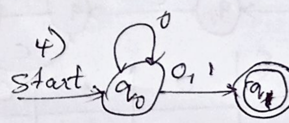
(NFA)

STT

$Q \backslash \Sigma$	0	1
$\rightarrow q_0$	q_1	q_0, q_1
q_1	q_0, q_1	q_1

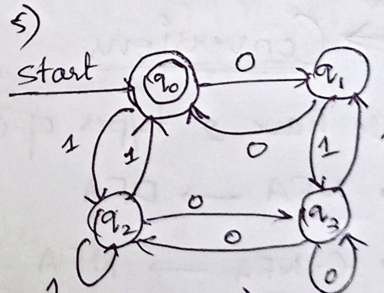
$\rightarrow 2^Q$

2^Q

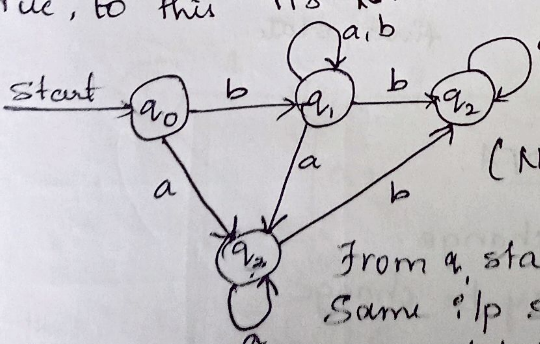
4) 

(NFA)

$Q \backslash \Sigma$	0	1
$\rightarrow q_0$	q_0, q_1	q_1
q_1	ϕ	ϕ

5) 

(NFA)

3) 

(NFA)

From q_1 state on same i/p symbol same state (q_1) & next state (q_3, q_2).

STT

$Q \backslash \Sigma$	a	b
$\rightarrow q_0$	q_3	q_1
q_1	q_1, q_3	q_1, q_2
q_2	q_2	q_2
q_3	q_3	q_2

$\rightarrow 2^Q = 2^{a,b}$

** → Con

- If there is no transition from 1 to another state on same i/p symbol then 'ϕ'

→ ε NFA

· Formal def

In case of ε NFA, there may be more than 1 ε transition from a state on ε i/p symbol.

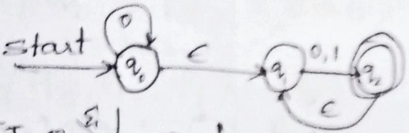
- ε

Formal def :

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

$$\boxed{\delta_{\epsilon NFA} = Q \times \Sigma \cup \{\epsilon\}}$$

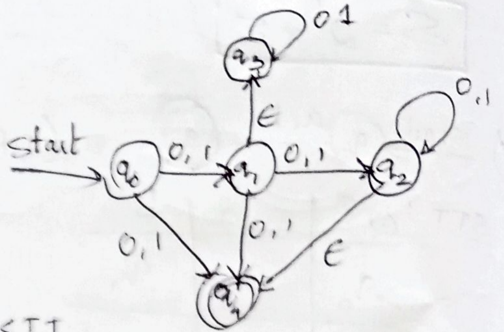
1.



STT

Q \ Σ	0	1	ε
→ q_0	q_0	ϕ	q_1
q_1	q_2	q_2	ϕ
q_2	ϕ	ϕ	q_1

2)



STT

Q \ Σ	0	1	ε
→ q_0	q_1, q_4	q_1, q_4	ϕ
q_1	q_2, q_4	q_2, q_4	q_3
q_2	q_2	q_2	q_4
q_3	q_3	q_3	ϕ
q_4	ϕ	ϕ	ϕ

*** → Conversions

We have 2 types of Conversion

- NFA ⟶ DFA
- ε-NFA ⟶ NFA

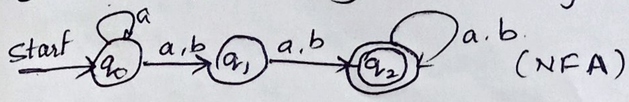
1) Conversion from NFA → DFA :

1. Starting state doesn't change.
2. Number of states may be change.
3. No of final state may be change.
4. In case of DFA, (q_0, q_1) & $(q_0\, q_1\, q_2)$ this states are consider as single state.

i.e. $(q_0\, q_1) \rightarrow$ 1 state, $(q_0\, q_1\, q_2) \rightarrow$ 1 state } DFA; — 2 state, — 3 state } NFA.

5. whenever NFA final state has appear in equivalent STT of DFA then we need to make it as final state.

1. Convert following NFA diagram into it's equivalent DFA.



→ Formal def of NFA

$Q \backslash \Sigma$	a	b
→ q_0	q_0, q_1	q_1
q_1	q_2	q_2
(q_2)	q_2	q_2

STT

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

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

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

$q_0 = \{q_0\}$; $F = \{q_2\}$.

Transition fun^n:

$\delta_{NFA} : Q \times \Sigma = 2^Q$

$\delta(q_0, a) \to q_0 q_1 = 2^Q$; $\delta(q_0, b) = q_1$

$\delta(q_1, a) = q_2$: $\delta(q_1, b) = q_2$

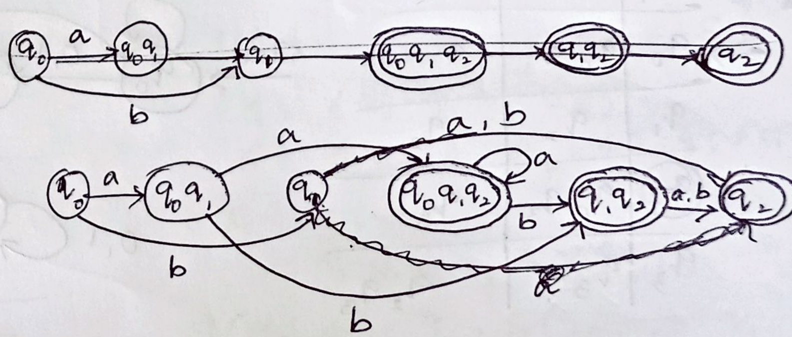
$\delta(q_2, a) = q_2$; $\delta(q_2, b) = q_2$.

Equivalent STT of DFA

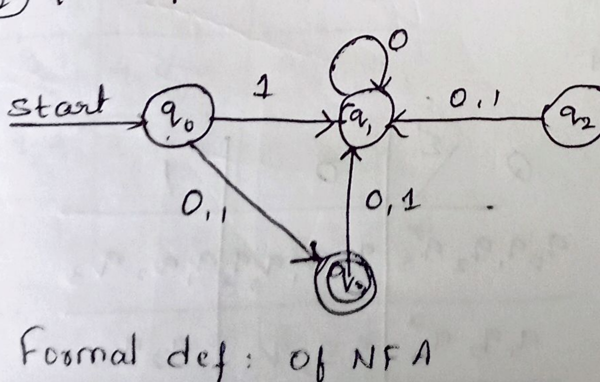
- The first row will be same as STT of NFA.

$Q \backslash \Sigma$	a	b
→ q_0	$q_0 q_1$	q_1
$q_0 q_1$	$q_0 q_1 q_2$	$q_1 q_2$
q_1	q_2	q_2
($q_0 q_1 q_2$)	$q_0 q_1 q_2$	q_1, q_2
($q_1 q_2$)	q_2	q_2
(q_2)	q_2	q_2

STD of DFA



2.



$Q \backslash \Sigma$	0	1
→ q_0	q_3	q_1, q_3
q_1	q_1	ϕ
q_2	q_1	q_1
(q_3)	q_1	q_1

→ Formal def: of NFA

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

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

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

$q_0 = \{q_0\}$; $F = \{q_3\}$

$\delta = Q \times \Sigma = 2^Q$

$= \{q_0\ q_1\ q_2\ q_3\} \times \{0, 1\}$

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

$\delta(q_1, 0) = q_1$; $\delta(q_1, 1) = \phi$

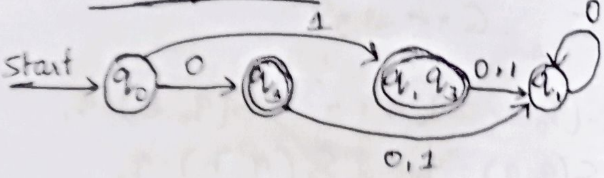
$\delta(q_2, 0) = q_1$; $\delta(q_2, 1) = q_1$

$\delta(q_3, 0) = q_1$; $\delta(q_3, 1) = q_1$

Equivalent STT of DFA

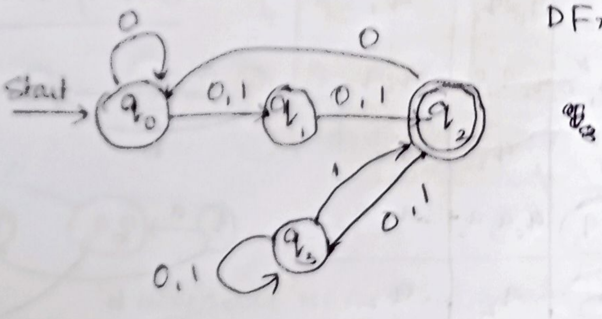
$Q \backslash \Sigma$	0	1
$\rightarrow q_0$	q_3	q_1, q_3
q_3	q_1	q_1
q_1, q_3	q_1	q_1
q_1	q_1	ϕ

STD of DFA



3. Convert following STT of NFA into its equivalent DFA

$Q \backslash \Sigma$	0	1
$\rightarrow q_0$	$q_0\ q_1$	q_1
q_1	q_2	q_2
q_2	$q_0\ q_3$	q_3
q_3	q_3	$q_2\ q_3$



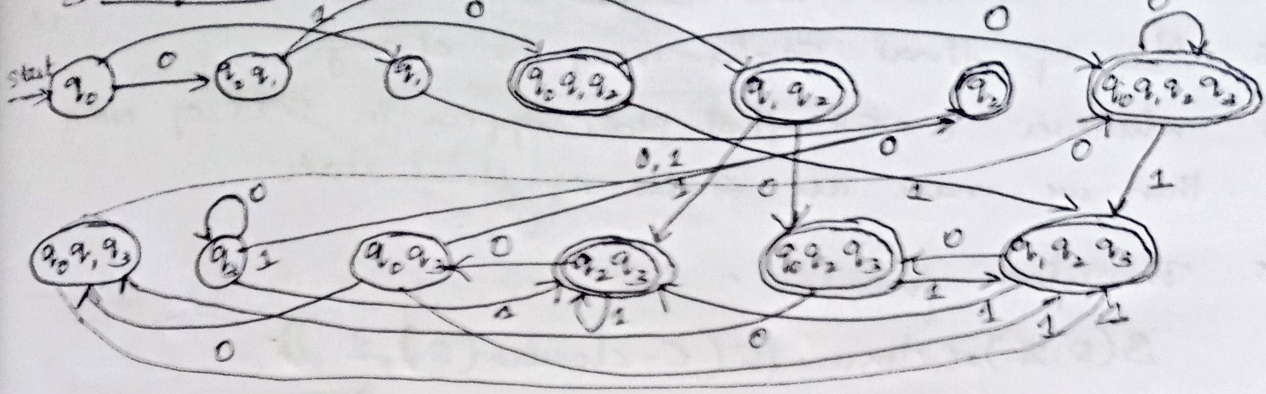
$\rightarrow$ Equivalent STT of DFA

$Q \backslash \Sigma$	0	1
$\rightarrow q_0$	$q_0\ q_1$	q_1
$q_0\ q_1$	$q_0\ q_1 q_2$	$q_1\ q_2$
q_1	q_2	q_2
$q_0 q_1 q_2^*$	$q_0 q_1 q_2 q_3$	$q_1 q_2 q_3$
$q_1 q_2^*$	$q_0 q_2 q_3$	$q_2 q_3$
q_2^*	$q_0\ q_3$	q_3

$Q \backslash \Sigma$	0	1
$q_0 q_1 q_2 q_3^*$	$q_0\ q_1 q_2 q_3$	$q_1 q_2 q_3$
$q_1 q_2 q_3^*$	$q_0 q_2 q_3$	$q_2 q_3$
$q_0\ q_2 q_3^*$	$q_0\ q_1 q_3$	$q_1 q_2 q_3$
$q_2\ q_3^*$	$q_0\ q_3$	$q_2 q_3$
$q_0\ q_3$	$q_0 q_1 q_3$	$q_1 q_2 q_3$
q_3	q_3	$q_2\ q_3$

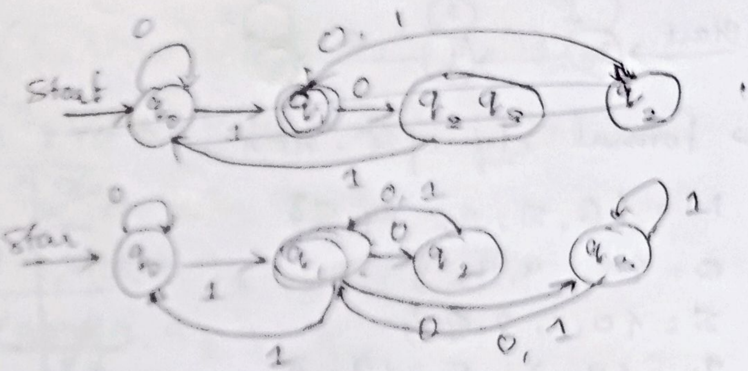
$q_0 q_1 q_3$

STD of DFA



Conversion of NFA to DFA:

Q \ Σ	0	1
→ q_0	q_0	q_1
q_1	$q_2 q_3$	q_0
q_2	q_1	q_1
q_3	q_1	$q_1 q_3$



→ Equivalent STT of DFA

Q \ Σ	0	1
→ q_0	q_0	q_1
q_1*	$q_2 q_3$	q_0
$q_2 q_3$	q_1	$q_1 q_3$
$q_1 q_3$*	$q_1 q_2 q_3$	$q_0 q_1 q_3$
$q_1 q_2 q_3$*	$q_1 q_2 q_3$	$q_0 q_1 q_3$
$q_0 q_1 q_3$*	$q_0 q_1 q_2 q_3$	$q_0 q_1 q_3$
$q_0 q_1 q_2 q_3$*	$q_0 q_1 q_2 q_3$	$q_0 q_1 q_3$
~~$q_0 q_1$~~	~~$q_0 q_2 q_3$~~	~~$q_0 q_1$~~

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

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

$\Sigma = \{0, 1\}$; $q_0 = \{q_0\}$; $F = \{q_1\}$

$\delta = Q \times \Sigma$

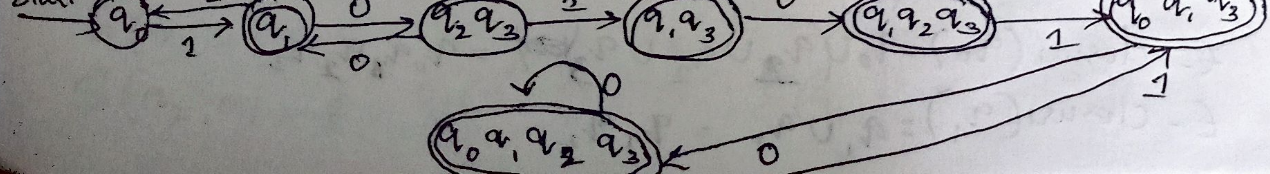
$= \{q_0\ q_1\ q_2\ q_3\} \times \{0, 1\}$

$\delta(q_0, 0) = q_0$; $\delta(q_0, 1) = q_1$

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

$\delta(q_2, 0) = q_1$; $\delta(q_2, 1) = q_1$

$\delta(q_3, 0) = q_1$; $\delta(q_3, 1) = q_1, q_3$



→ ε-NFA to NFA:

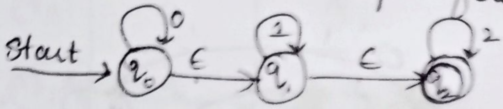
1. When we convert initial state doesn't change.
2. Number of states must be ~~the~~ same.
3. No of final states may be change.
4. whenever ε NFA final state appear in STT of NFA then we make all states as final state.
5. Transition formula:

$$\delta(Q, \Sigma) = \epsilon\text{-closure}(\delta(\epsilon\text{-closure}(Q), \Sigma))$$

* ϵ-closure$(\phi) = \phi$

---

1. Find ε-closures for following ε-NFA.



→ Formal def of ε-NFA:

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

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

$\Sigma = \{0, 1, 2, \epsilon\}$

$q_0 = \{q_0\}$; $F = \{q_2\}$

STT of ε-NFA:

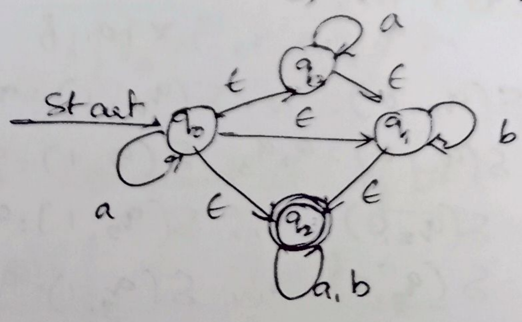
Q \ Σ	0	1	2	ε
→ q_0	q_0	ϕ	ϕ	q_1
q_1	ϕ	q_1	ϕ	q_2
q_2*	ϕ	ϕ	q_2	ϕ

ε closure $(q_0) = q_0 \cup q_1 \cup q_2 \cup \phi = q_0 q_1 q_2$

ε-closure $(q_1) = q_1 \cup q_2 \cup \phi = q_1 q_2$

ε-closure $(q_2) = q_2 \cup \phi = q_2$

2.



STT of ε-NFA

Q \ Σ	a	b	ε
→ q_0	q_0	ϕ	$q_1 q_2 q_3$
q_1	ϕ	q_1	q_2
q_2*	q_2	q_2	ϕ
q_3	q_3	ϕ	q_1

→ Formal def of ε-NFA

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

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

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

$q_0 = \{q_0\}$ $F = \{q_2\}$

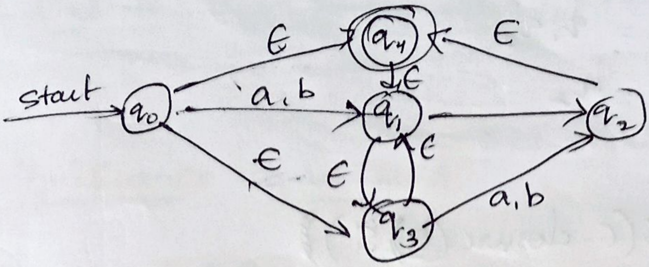
ε-closure $(q_0) = q_0 \cup (q_1 \cup q_2 \cup q_3) \cup \phi = q_0 q_1 q_2 q_3$

ε-closure $(q_1) = q_1 \cup q_2 = q_1 q_2$

ϵ-closure$(q_2) = q_2 \cup \phi = q_2$

ϵ-closure$(q_3) = q_3 \cup (q_1 \cup q_2) = q_1 q_2 q_3$

3.



$\rightarrow$ $M = \{Q, \Sigma', q_0, F, \delta\}$

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

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

$q_0 = \{q_0\}$; $F = \{q_4\}$

ϵ-closure$(q_0) = q_0 \cup (q_3 \cup q_4) \cup q_1 \cup q_2 = q_0 q_1 q_2 q_3 q_4$

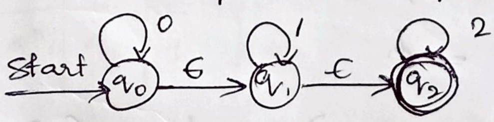
ϵ closure$(q_1) = q_1 \cup (q_2 \cup q_3) \cup q_4 = q_1 q_2 q_3 q_4$

ϵ closure$(q_2) = q_2 \cup q_4 \cup q_1 \cup q_3 = q_1 q_2 q_3 q_4$

ϵ closure$(q_3) = q_3 \cup q_1 \cup q_2 \cup q_4 = q_1 q_2 q_3 q_4$

ϵ closure$(q_4) = q_4 \cup q_1 \cup q_2 \cup q_3 = q_1 q_2 q_3 q_4$

$\rightarrow$ ϵ-NFA diagram to Equivalent NFA diagram



$\rightarrow$ Formal def:

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

$Q: \{q_0 q_1 q_2\}$

$\Sigma: \{0, 1, 2, \epsilon\}$

$q_0: \{q_0\}$; $F = \{q_2\}$

STT of ϵ NFA

$Q \backslash \Sigma$	0	1	2	ϵ
$\rightarrow q_0$	q_0	ϕ	ϕ	q_1
q_1	ϕ	q_1	ϕ	q_2
$*q_2$	ϕ	ϕ	q_2	ϕ

Transition fun of ϵ-NFA

$$\boxed{\delta_{\epsilon NFA} : Q \times \Sigma' \cup (\epsilon) \rightarrow 2^Q}$$

$= \{q_0 q_1 q_2\} \times \{0, 1, 2\} \cup \{\epsilon\}$

$\delta(q_0, 0): q_0$; $\delta(q_0, 1): \phi$; $\delta(q_0, 2): \phi$; $\delta(q_0, \epsilon): q_1$

$\delta(q_1, 0) = \phi$; $\delta(q_1, 1) = q_1$; $\delta(q_1, 2)\ \phi$; $\delta(q_1, \epsilon): q_2$

$\delta(q_2, 0): \phi$; $\delta(q_2, 1) = \phi$; $\delta(q_2, 2): q_2$ $\delta(q_2, \epsilon): \phi$

ϵ- closures

ϵ-closure$(q_0) = q_0 \cup q_1 \cup q_2 = q_0 q_1 q_2$

ϵ-closure$(q_1) = q_1 \cup q_2 = q_1 q_2$

ϵ-closure$(q_2) = q_2 \cup \phi = q_2$

Conversion

$\delta(Q, \Sigma) \to \epsilon$-closure $(\delta(\epsilon$-closure $(Q), \Sigma))$

$\delta(q_0, 0) \to \epsilon$closure $(\delta(q_0, q_1, q_2)\ 0)$

~~$\delta(q_0, 1)$~~ ~~$\delta(q_0 2)$~~ ϵ closure $(\delta(q_0, 0) \cup \delta(q_1, 0) \cup \delta(q_2, 0))$

ϵ closure $(q_0 \cup \phi \cup \phi)$

ϵ closure $(q_0) \Rightarrow q_0 q_1 q_2$

$\delta(q_0, 1) - \epsilon$ closure $(\delta(q_0, q_1, q_2) \cdot 1)$

ϵ closure $(q_1) = q_1 q_2$

$\delta(q_0, 2) - \epsilon$closure $(\delta(q_0 q_1 q_2) 2)$

ϵclosure $(q_2) = q_2$

$\delta(q_1, 0) \to \epsilon$ closure $(\delta(q_1 q_2) 0)$

ϵ closure $(\phi) = \phi$

$\delta(q_1, 1) \Rightarrow \epsilon$ closure $(\delta(q_1, q_2)\ 1) \Rightarrow \epsilon$ closure(q_1)

$= q_1 q_2$

$\delta(q_1, 2) \to \epsilon$ closure$(\delta(q_1, q_2) 2) \Rightarrow \epsilon$closure$(q_2) = q_2$

$\delta(q_2, 0) \to \epsilon$ closure $(q_2) 0) \Rightarrow \epsilon$ closure$(\phi) = \phi$

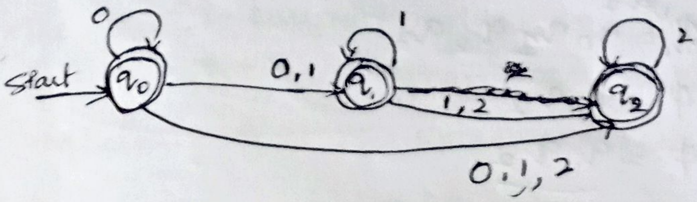
$\delta(q_2, 1) \to \epsilon$ closure $(\phi) \Rightarrow$ ~~q_2~~ ϕ

$\delta(q_2, 2) \Rightarrow \epsilon$ closure $(q_2) = q_2$

STT of NFA

$Q \backslash \Sigma$	0	1	2
$\to q_0{*}$	$q_0 q_1 q_2$	$q_1 q_2$	q_2
q_1^*	ϕ	q_1, q_2	q_2
q_2^*	ϕ	ϕ	q_2

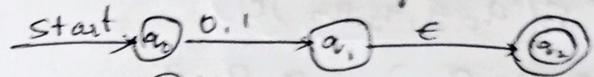
STD of NFA



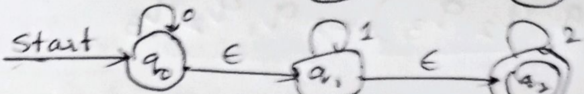
→ Significance of ϵ-NFA

1. Without receiving any input symbol from 1 state to another state finally it will reach the final state.

eg:



$\Rightarrow 0 1 \epsilon = 0 1$

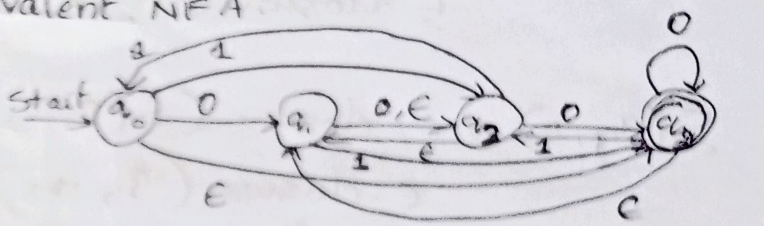


$\rightarrow 0 \epsilon 1 \epsilon 2 = 0 1 2$

$0 0 \epsilon 1 1 \epsilon 2 2 \Rightarrow 0 0 1 1 2 2$.

2. ϵ-NFS STT to equivalent NFA

Q \ Σ	0	1	ϵ
→ q_0	q_1	q_2	q_3
q_1	q_2	q_3	q_2
q_2	q_3	q_0	q_1
q_3^*	q_3	q_2	q_1



→ Formal def:

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

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

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

$q_0 = \{q_0\}$; $F = \{q_3\}$

Transition of ϵ-NFA

$\delta_{\epsilon\text{-NFA}} = Q \times \Sigma \cup \{\epsilon\} \rightarrow 2^Q$

$\{q_0, q_1, q_2, q_3\} \times \{0, 1\} \cup \{\epsilon\}$

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

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

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

ϵ- closure

ϵ-closure$(q_0) = q_0 \cup q_3 \cup q_1 = q_0 q_1 q_2 q_3$

ϵ-closure$(q_1) = q_1 \cup q_2 \cup \phi = q_1 q_2$

ϵ-closure$(q_2) = q_2 \cup q_1 \cup \phi = q_1 q_2$

ϵ-closure$(q_3) = q_3 \cup q_1 \cup q_2 \cup \phi = q_1 q_2 q_3$.

Conversion

$\delta(Q, \Sigma) \rightarrow \epsilon$-closure$[\delta(\epsilon$- closure$(Q)\ \Sigma_i)]$

$\delta(q_0, 0) \rightarrow \epsilon$-closure $(\delta(\overset{\epsilon\text{-closure}}{q_0 q_1 q_2 q_3})0)$

ϵ-closure $(q_1 q_2 q_3) \Rightarrow q_1 q_2 q_3$

$\delta(q_0, 1) \rightarrow \epsilon$-closure$(\delta(q_0 q_1 q_2 q_3)1)$

ϵ-closure $(q_0 q_2 q_3) \Rightarrow q_0 q_1 q_2 q_3$

$\delta(q_1, 0) \rightarrow \epsilon$-closure $(\delta(q_1 q_2 q_3)0)$

ϵ-closure $(q_2 q_3) \Rightarrow q_1 q_2 q_3$

$\delta(q_1, 1) \rightarrow \epsilon$-closure$(\delta(q_1 q_2 q_3)1)$

ϵ-closure $(q_0 q_2 q_3) \Rightarrow q_0 q_1 q_2 q_3$.

$\delta(q_2, 0) \rightarrow \epsilon$-closure $(\delta(q_2 q_3)0)$

ϵ-closure $(q_3) \Rightarrow q_1 q_2 q_3$

$\delta(q_2, 1) \rightarrow \epsilon$-closure$(\delta(q_2 q_3)1)$

ϵ-closure $(q_0 q_2) \Rightarrow q_0 q_1 q_2 q_3$

$\delta(q_3, 0) \rightarrow \epsilon$-closure $(\delta(q_3)0)$

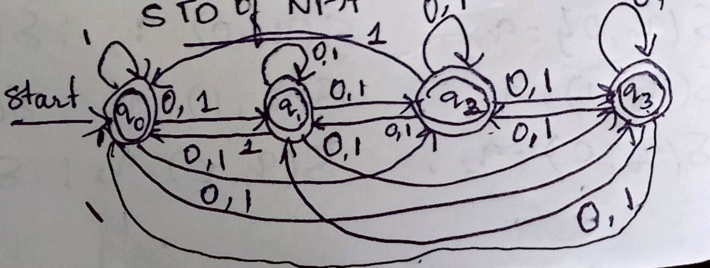
ϵ-closure $(q_3) \Rightarrow q_1 q_2 q_3$

$\delta(q_3, 1) \rightarrow \epsilon$closure$(q_2) = q_1 q_2 q_3$

STT of NFA

$Q \setminus \Sigma$	0	1
$\rightarrow * q_0$	$q_1 q_2 q_3$	$q_0 q_1 q_2 q_3$
$* q_1$	$q_1 q_2 q_3$	$q_0 q_1 q_2 q_3$
$* q_2$	$q_1 q_2 q_3$	$q_0 q_1 q_2 q_3$
$* q_3$	$q_1 q_2 q_3$	$q_0 q_1 q_2 q_3$

STD of NFA



Minimization of Final State Machine: or Finate Automata

- The main purpose of FSM is to reduce the no of states in a given diagram as much as possible.
- Minimization of FSM prefer only less no of state.

Advantage:

1. By doing the mini of FSM Space complexity will be very less.
2. Time Complex (Chess)

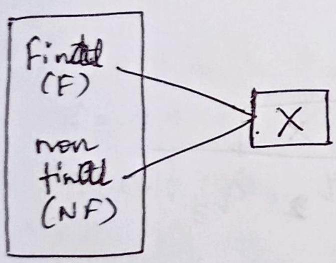
NOTE: Mini of FSM will be applicable to DFA's only (not of NFA & ϵ-NFA)

Construction of FSM:

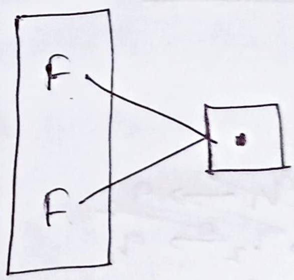
Minimization is depends on 2 constrains:

1. Distinguistable pair. (different)
2. Indistinguistable pair.

① Distinguistable pair:



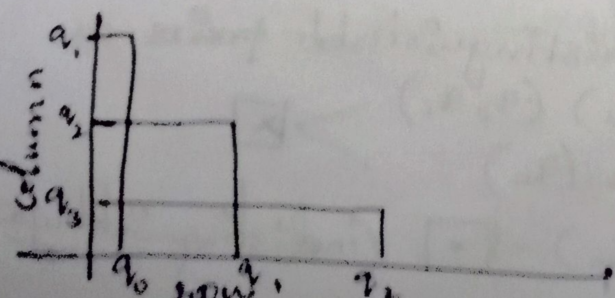
② Indistinguistable pair



minimization table can be constructed by taking (n-1) rows & (n-1) columns.

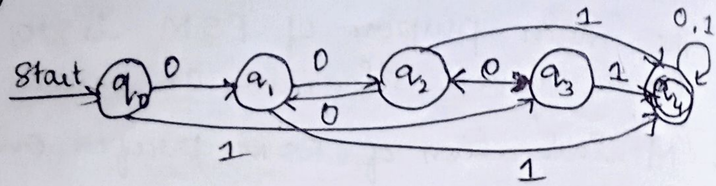
- In mini table while specifing column wise no of state we need to discard starting state (q_0)
- while specifing row wise no of state we need to discard highest no of state.

Eg: $Q = \{q_0, q_1, q_2, q_3\}$ n = no of state (4), starting state (q_0), highest no of state (q_3)



Q) Mini the following FSM

Q \ Σ	0	1
→ 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



Transition fun

① Formal defi . DFA

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

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

$q_0 = \{q_0\}$; $F = \{q_4\}$.

② $\delta = Q \times \Sigma$

$= \{q_0, q_1, q_2, q_3, q_4\} \times \{0, 1\}$

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

$\delta(q_0, 1) = q_3$; $\delta(q_1, 1) = q_4$; $\delta(q_2, 1) = q_4$; $\delta(q_3, 1) = q_4$

$\delta(q_4, 0) = q_4$

$\delta(q_4, 1) = q_4$

③ Minimization table (DFA)

row				
q_1	X			
q_2	X	•		
q_3	X	•	•	
q_4	X	X	X	X
	q_0	q_1	q_2	q_3

column

q_0 q_3

NF NF

$\cup\ q_0 \xrightarrow{0} q_1 \xrightarrow{0} q_2$

$q_3 \xrightarrow{0} q_2 \xrightarrow{0} q_1$

$q_0 q_3 \xrightarrow{0} (q_0, 0) \cup (q_3, 0) = q_1, q_2$; $q_2 q_3 \xrightarrow{0} q_1, q_2$

$q_0 q_3 \xrightarrow{1} (q_3\, q_4)$ (x) ; $q_2 q_3 \xrightarrow{1} q_4 q_4$ (•)

$q_1 q_3 \xrightarrow{0} (q_2 q_2)$; $q_1, q_2 \xrightarrow{0} q_1, q_2$

$q_1, q_3 \xrightarrow{1} (q_4 q_4)$ (•) ; $q_1, q_2 \xrightarrow{1} q_4 q_4$ (•)

$q_0 q_2 \xrightarrow{0} q_1, q_1$

$q_0 q_1 \xrightarrow{0} q_1, q_2$; $q_0 q_2 \xrightarrow{1} q_3 q_4$

$q_0 q_1 \xrightarrow{1} q_3 q_4$.

④ From the above table distinguistable pairs are

(q_0, q_4) (q_0, q_3) $(q_0 q_2)$ $(q_0 q_1)$ > $\boxed{X}$

(q_1, q_4) $(q_2 q_4)$ $(q_3 (q_4)$.

(q_1, q_3) $(q_2 q_3)$ (q_1, q_2) — $\boxed{\bullet}$ (indistinguistable pair)

$q_1 = q_2 = q_3$

All indistinguistable pairs can be merged with small state (single state)

(8)

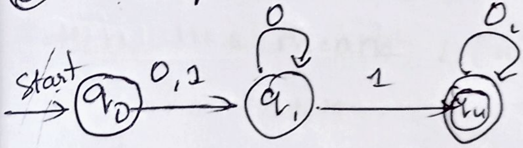
$Q \setminus \Sigma$	0	1
$\rightarrow q_0$	q_1	~~q_3~~ q_1
q_1	~~q_2~~ q_1	q_4
q_2	q_1	q_4
q_3	q_2	q_4
q_4^*	q_4	q_4

mini STT of DFA

$Q \setminus \Sigma$	0	1
$\rightarrow q_0$	q_1	q_1
q_1	q_1	q_4
q_4^*	q_4	q_4

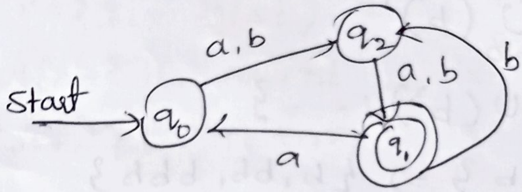
we can't merge all distinct Pairs.

(6) STD of mini DFA:



- As per mini of rule we can able to merge all indist pair with single state.

Q)



$Q \setminus \Sigma$	a	b
$\rightarrow q_0$	q_2	q_2
q_1	q_0	q_2
q_2^*	q_1	q_1

$\rightarrow Q = \{q_0\ q_1\ q_2\ \cancel{q_3}\}$

$\Sigma = \{a, b\}$; $q_0 = \{q_0\}$; $F = \{q_1\}$

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

$\delta(q_0, b) = q_2$; $\delta(q_1, b) = q_2$; $\delta(q_2, b) = q_1$

	q_0	q_1
q_1	X	
q_2	X	X

$(q_0\ q_2) \xrightarrow{0} q_1\ q_2$
F NF

$(q_1, q_2) \xrightarrow{0} q_1 q_0$

If all $\boxed{X}$ minization of FSM is not possible.

If all $\boxed{\cdot}$ the merging. we need to be