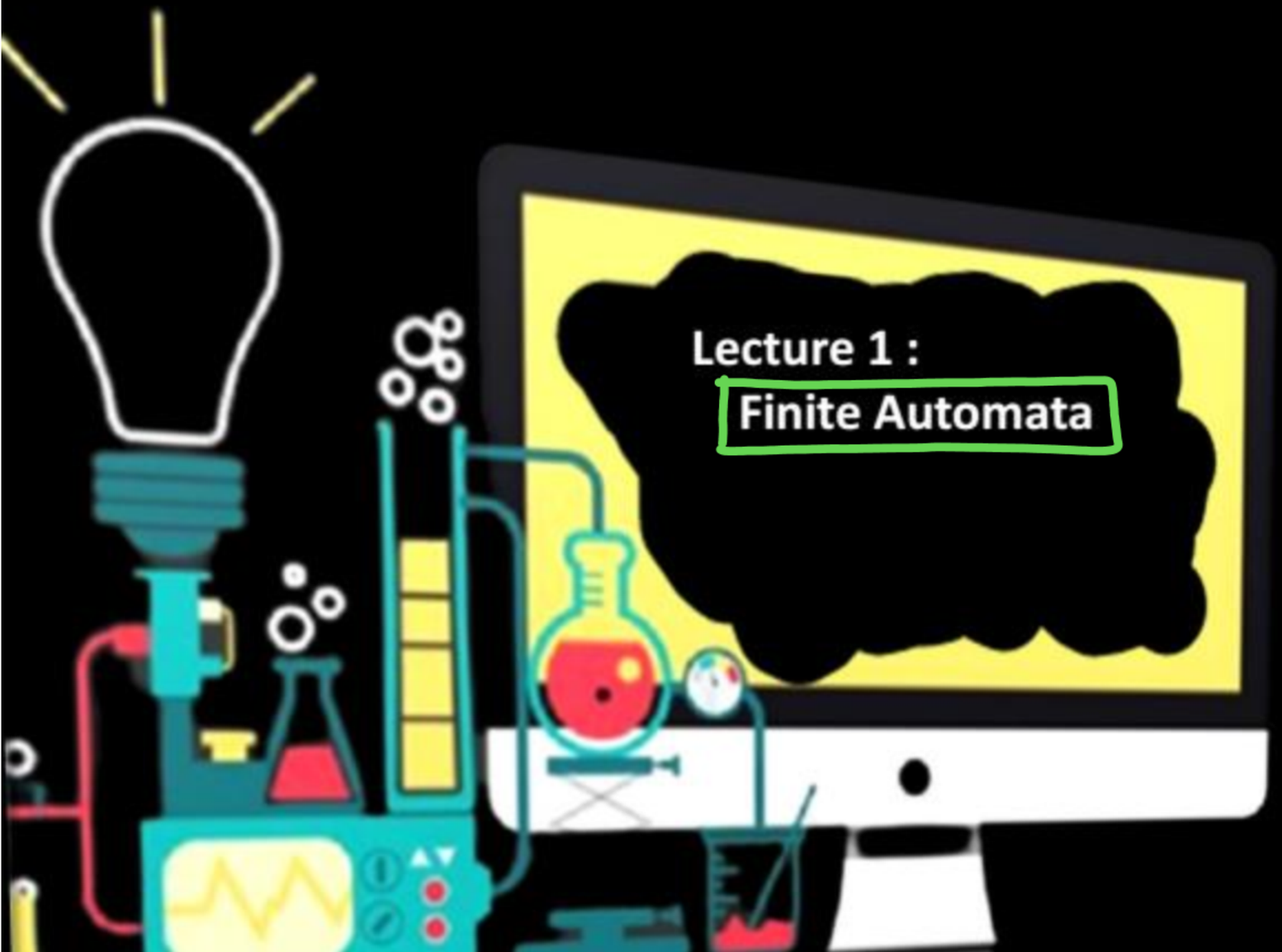
CS & IT Engineering

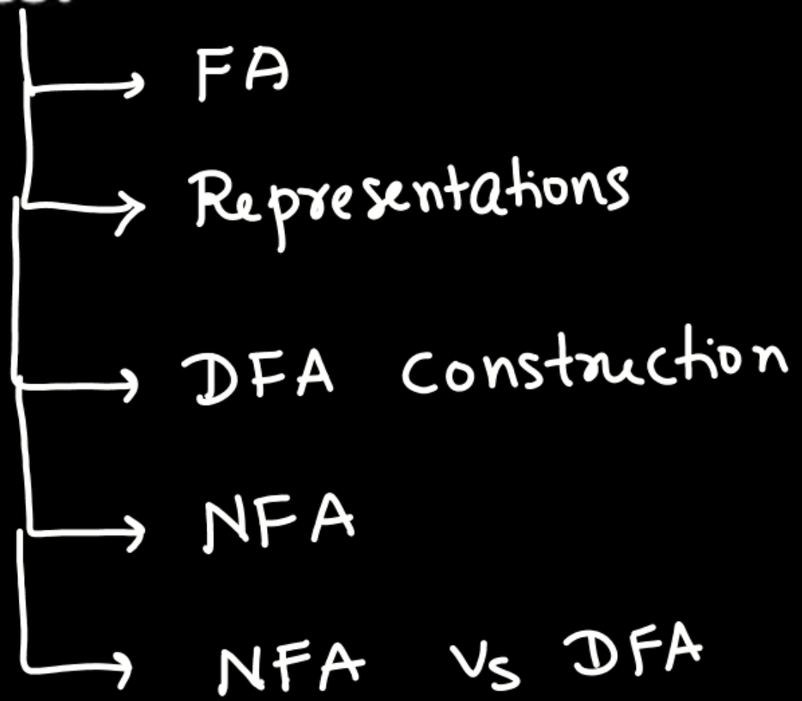


Lecture 1 :

Finite Automata

Deva sir

Topics: To be covered



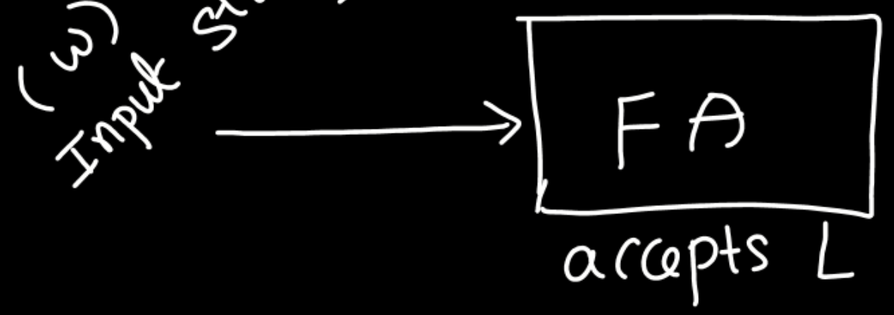
What is FA ?

Finite Automata (FA) / Finite Machine (FM) / Finite state machine (FSM)

- It is a m/c
- It represents a regular language
- It accepts/recognizes a regular set.

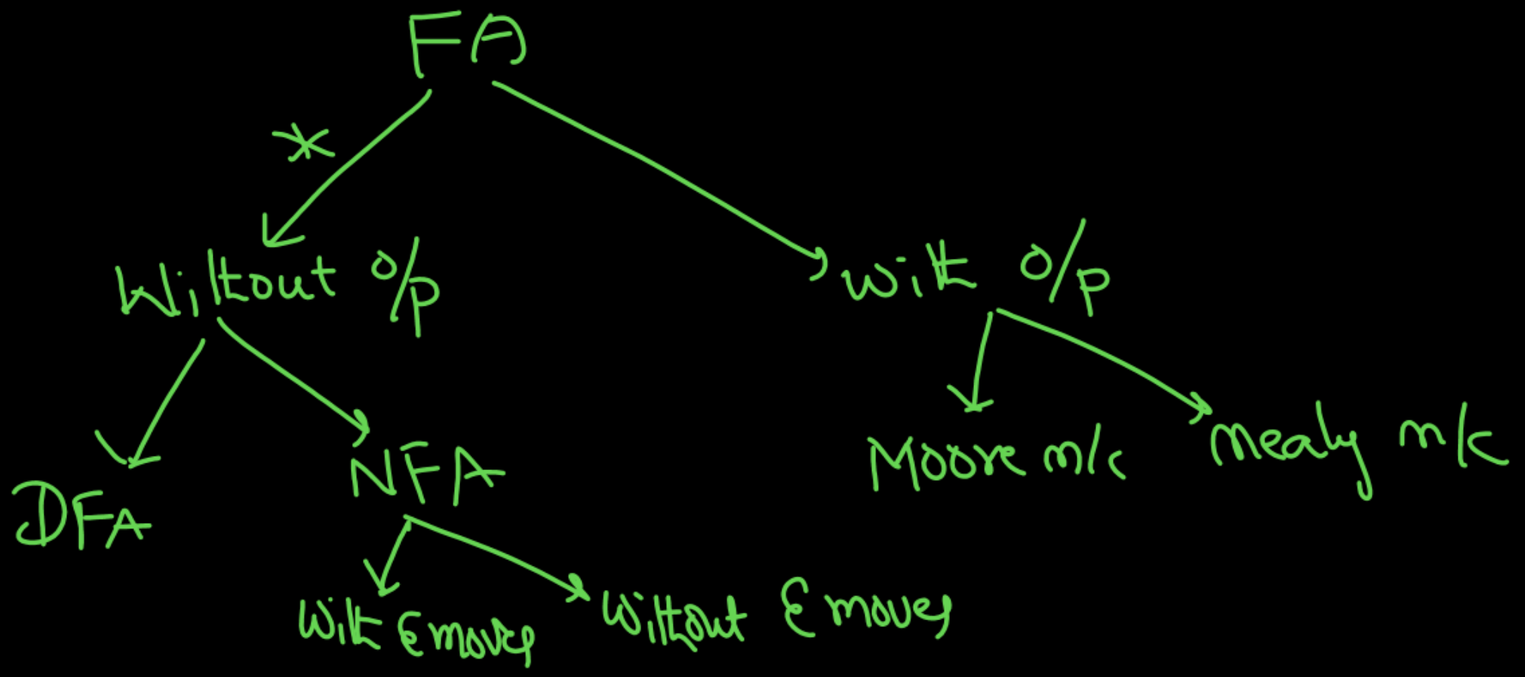
FA ≅ Regular Set

Language accepted by FA is Regular language

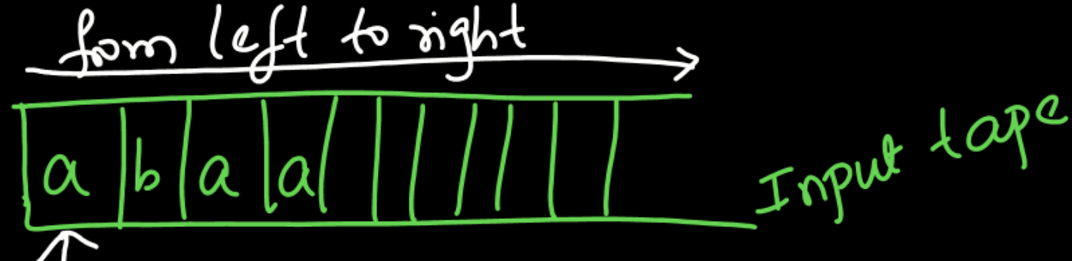


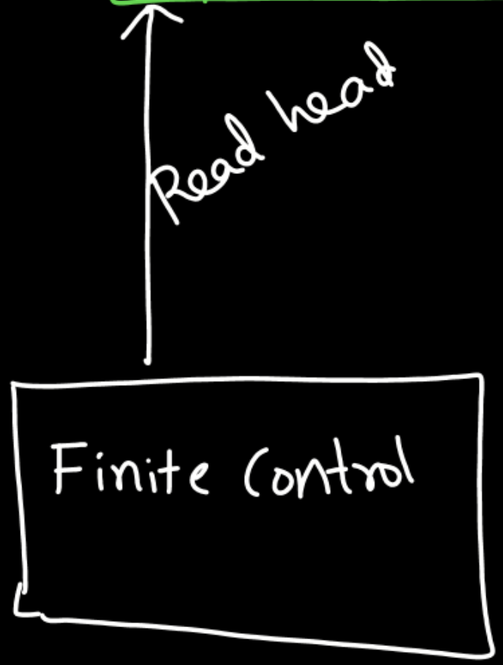
If $w \in L$, w is accepted by FA (valid string)

If $w \notin L$, w is not accepted by FA (invalid string)



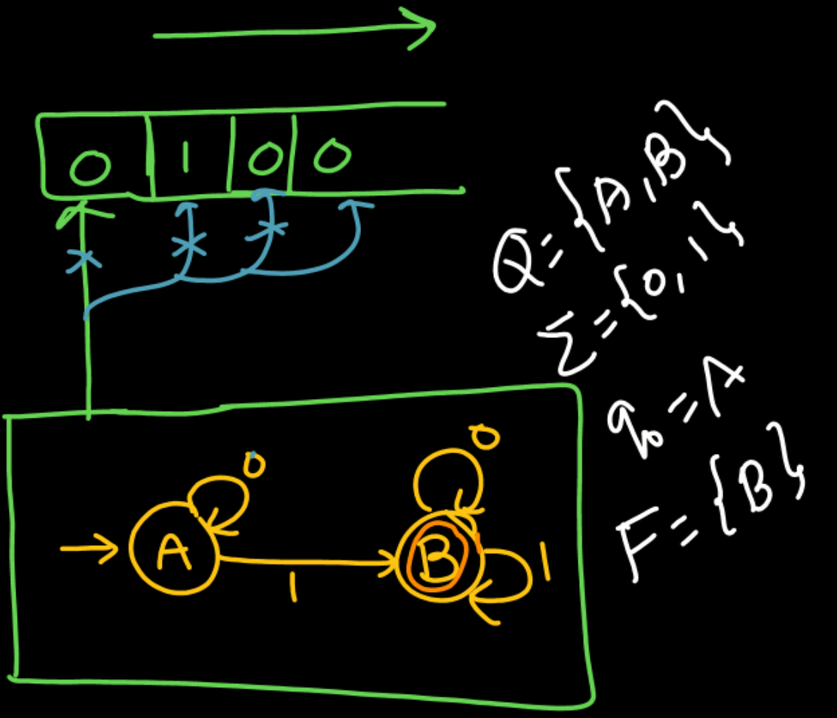
Finite Automata configuration :





$FA = (Q, \Sigma, \delta, q_0, F)$

A is initial
B is final
A is non-final



A $\xrightarrow{0}$ A $\xrightarrow{1}$ B $\xrightarrow{0}$ B $\xrightarrow{0}$ B

A: Start State

B: Halting state

$FA = (Q, \Sigma, \delta, q_0, F)$

- Q → Set of states
- Σ → Set of input symbols
- δ → Transition Function
- q_0 → Initial state $q_0 \in Q$
- F → Set of final states $F \subseteq Q$

Deterministic Finite Automata (DFA):

$$\delta : \underbrace{Q \times \Sigma}_{\text{Domain}} \rightarrow \underbrace{Q}_{\text{co-domain}}$$

Non-deterministic Finite Automata (NFA):

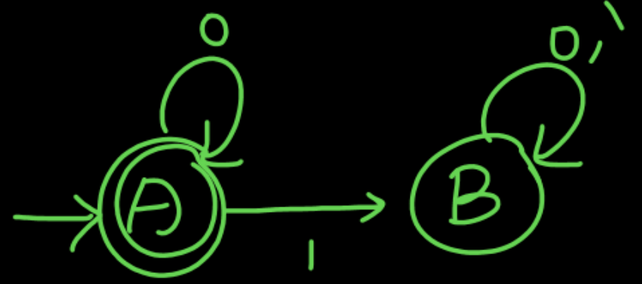
1) NFA without ε moves

$$\delta : \underbrace{Q \times \Sigma}_{\text{Domain}} \rightarrow \underbrace{2^Q}_{\text{co-domain}}$$

2) NFA with ε moves

$$\delta : \underbrace{Q \times \Sigma \cup \{\varepsilon\}}_{\text{Domain}} \rightarrow \underbrace{2^Q}_{\text{co-domain}}$$

DFA



$\delta(A,0) = A \in Q$

$\delta(A,1) = B$

$\delta(B,0) = B$

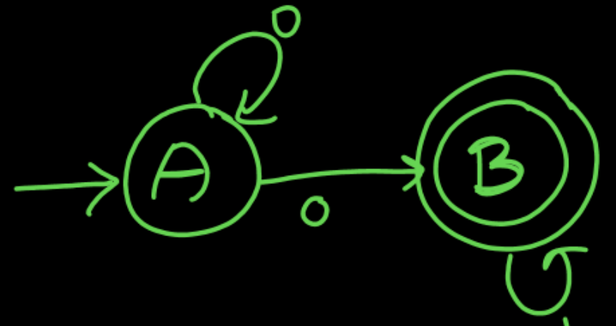
$\delta(B,1) = B$

$\delta : Q \times \Sigma \to Q$

$\delta = \{((A,0), A), \ldots\}$

NFA without ε-moves

$\delta : Q \times \Sigma \to 2^Q$



$\delta(A,0) = \{A,B\} \in 2^Q$

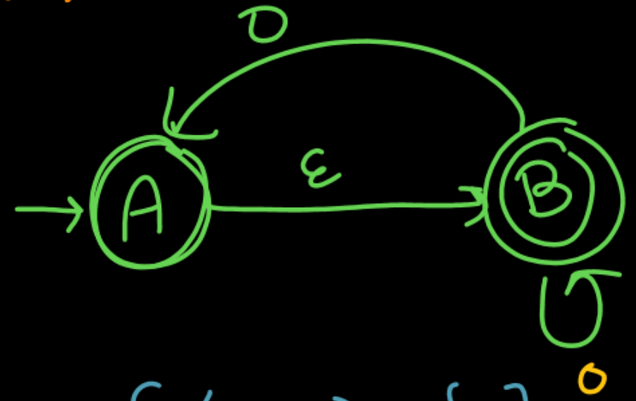
$\delta(A,1) = \{\}$

$\delta(B,0) = \{\}$

$\delta(B,1) = \{B\}$

NFA with ε moves

$\delta : Q \times \Sigma \cup \{\varepsilon\} \to 2^Q$



$\delta(A,\varepsilon) = \{B\}$

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

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

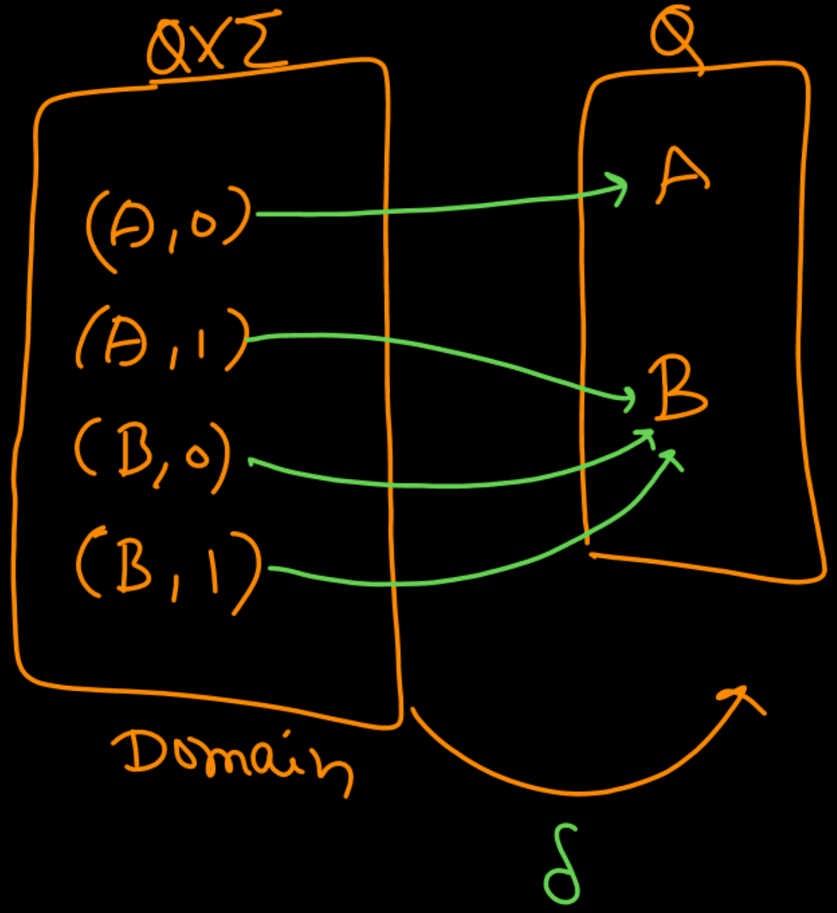
$\delta(B,0) = \{A,B\}$

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

$\delta(B,\varepsilon) = \phi$

DFA

$\delta : Q \times \Sigma \rightarrow Q$



$Q = \{A, B\}$

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

$Q \times \Sigma = \{(A,0), (A,1), (B,0), (B,1)\}$

Function :

$f : A \rightarrow B$

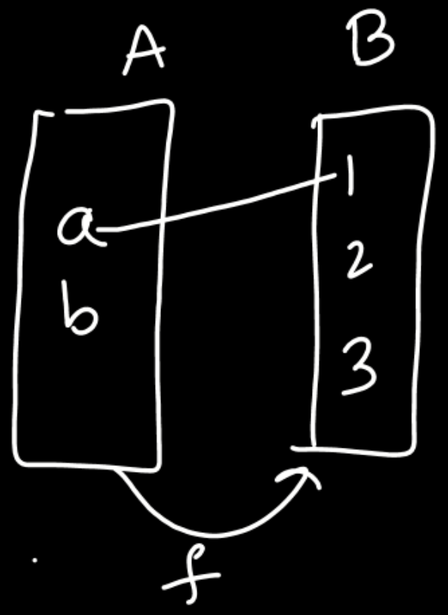
A : Domain, B : co-domain

I) Relation $f \subseteq A \times B$

II) Existence : Every element of A will have image in B.

III) Uniqueness : every element of A will have Single image in B.

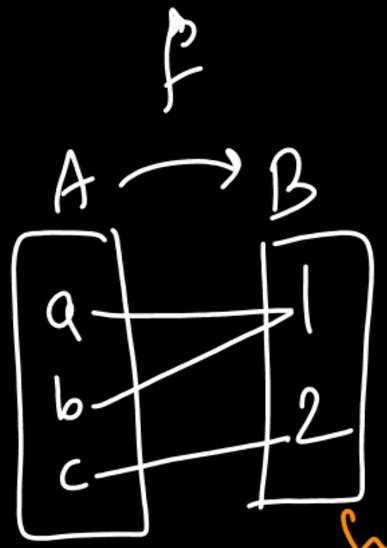
f is a function
||
f is a relation
||
f is a set



$f(b) = ?$

not function

$f = \{(a,1)\}$

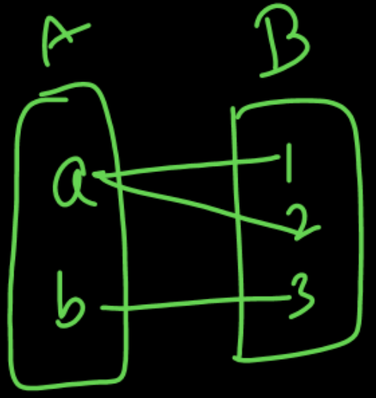


function

$f = \{(a,1), (b,1), (c,2)\}$

$= \{(a,\_), (b,\_), (c,\_)\}$

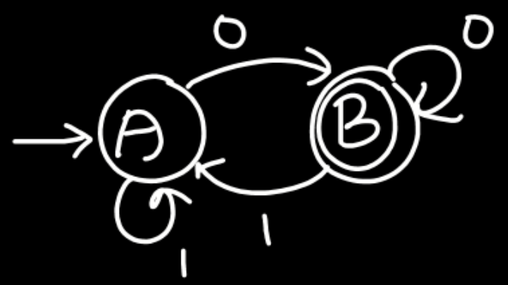
$|f| = |A|$



not function

Finite Automata Representations :

① State Diagram



$A \xrightarrow{1} A$

$\delta(A,1) = A$

$((A,1),A) \in \delta$

A is initial state

B is final state

Final states ⇉ B
Non finals ⇉ A

② Transition Table

δ	0	1
→ A	B	A
(B) OR * B	B	A

$\delta : Q \times \Sigma \to Q$

$\delta = \{ (\delta(A,0)=B) \to ((A,0),B),\ (\delta(A,1)=A) \to ((A,1),A),\ (\delta(B,0)=B) \to ((B,0),B),\ (\delta(B,1)=B) \to ((B,1),B) \}$

③ Set

A is initial

B is final

$\delta(A,0) = B$

$((A,0),B) \in \delta$

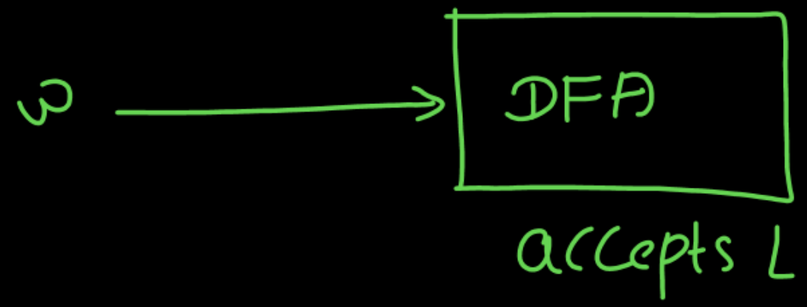
$\delta = \{((A,0),B), \ldots\}$

$f(x) = y$

$(x,y) \in f$

$f = \{(x,y), \ldots\}$

What is DFA ?

I) 

If $w \in L$, After reading whole string DFA halts at final state

If $w \notin L$, DFA halts at nonfinal state

II) $\delta : Q \times \Sigma \to Q$

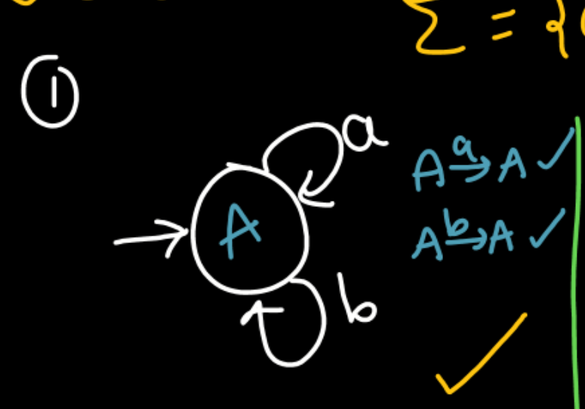
III) From every state in Q, for every symbol in Σ ($Q \times \Sigma$),

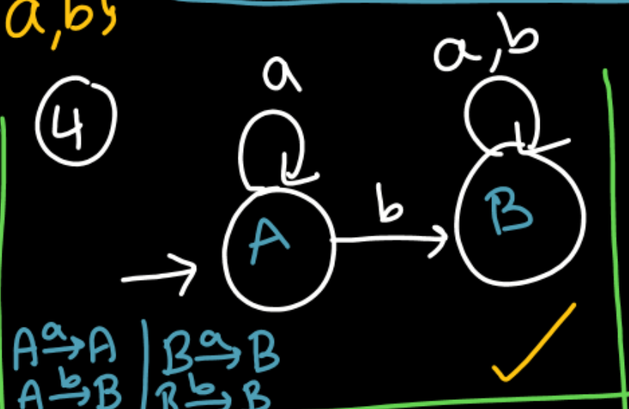
DFA contain exactly one transition to next state in Q. (Q)

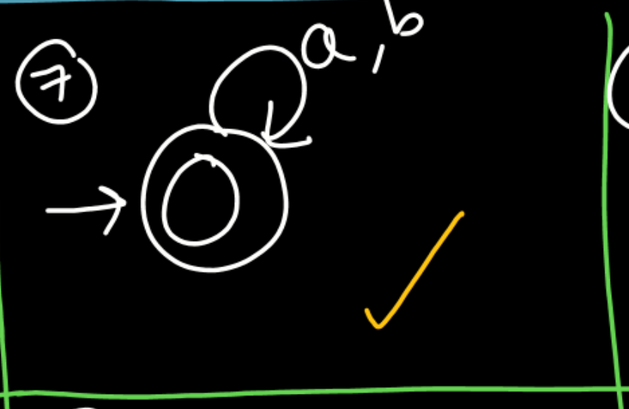
Identify DFAs:

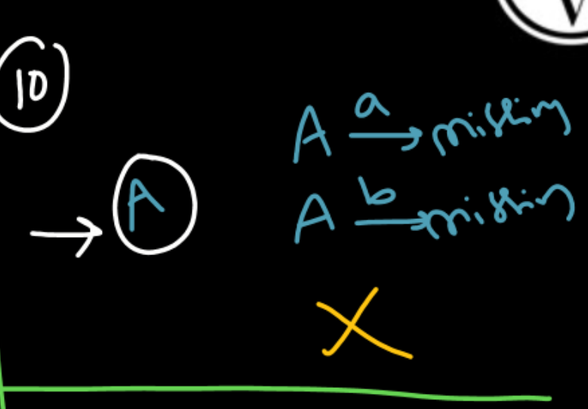
$\Sigma = \{a, b\}$

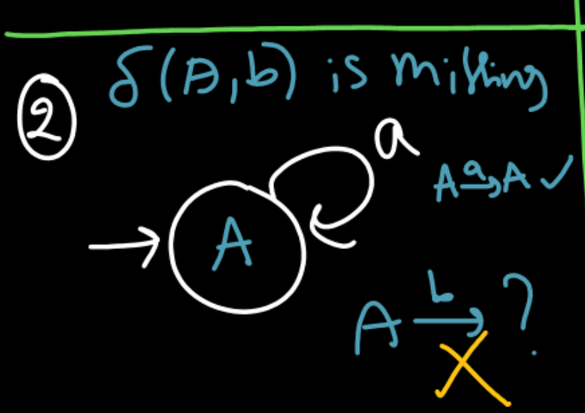
From each state, for every i/p symbol, exactly one transition

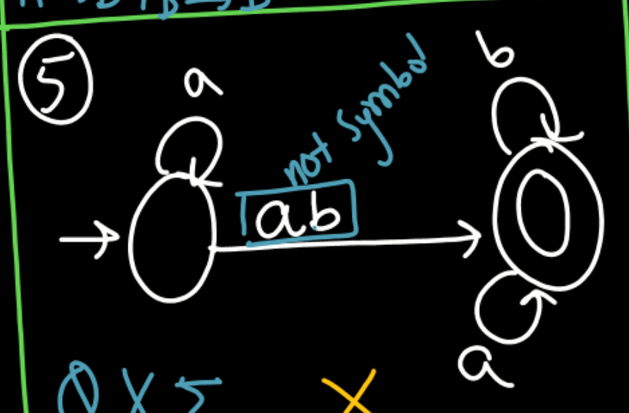


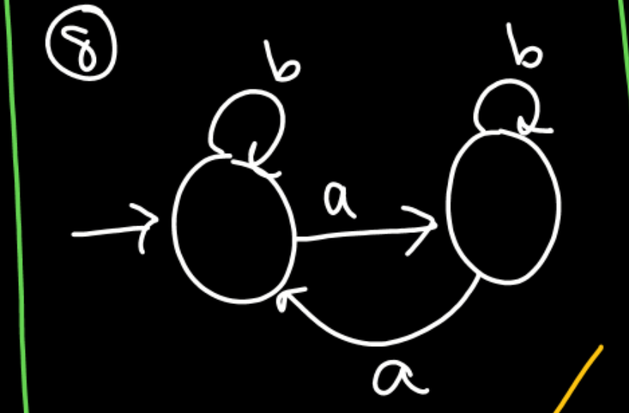


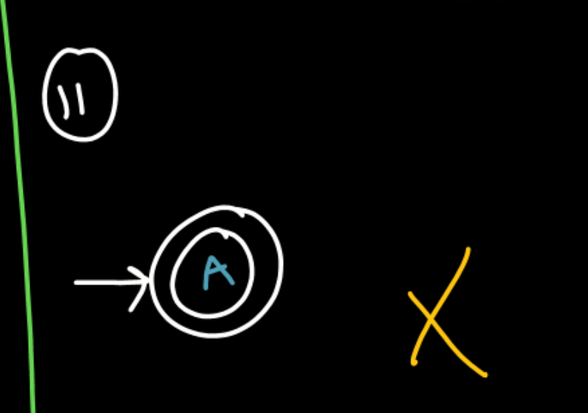


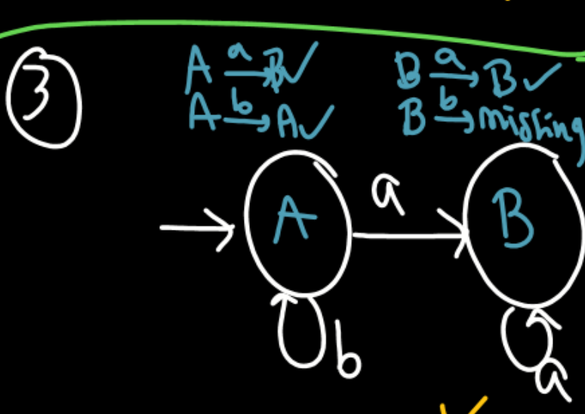


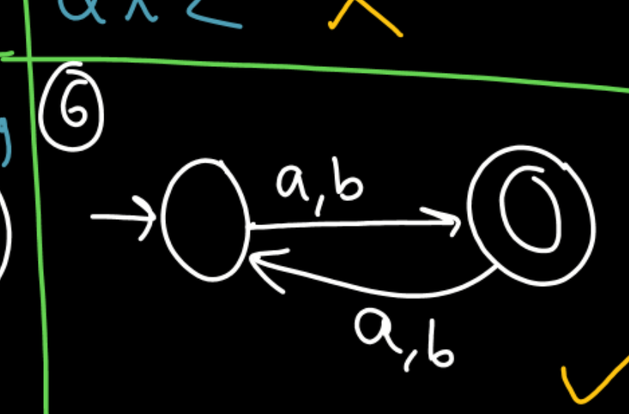


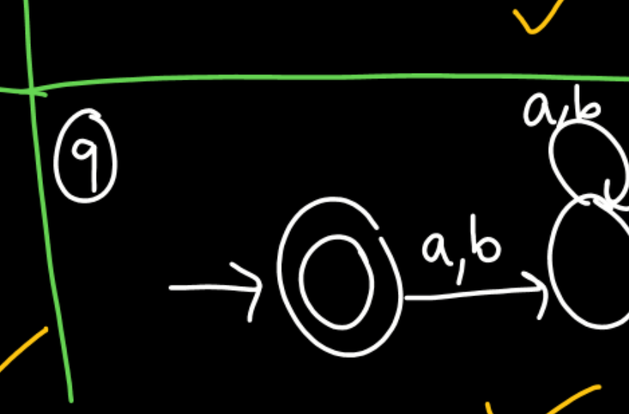


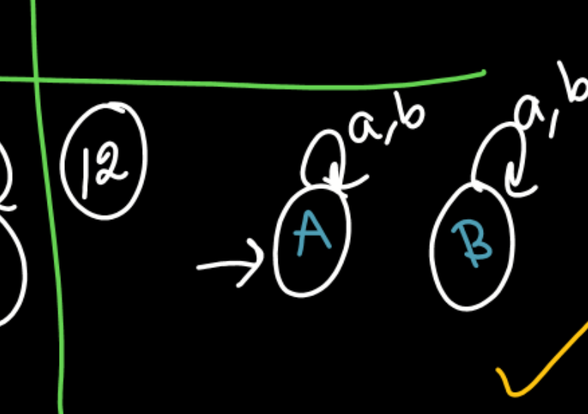


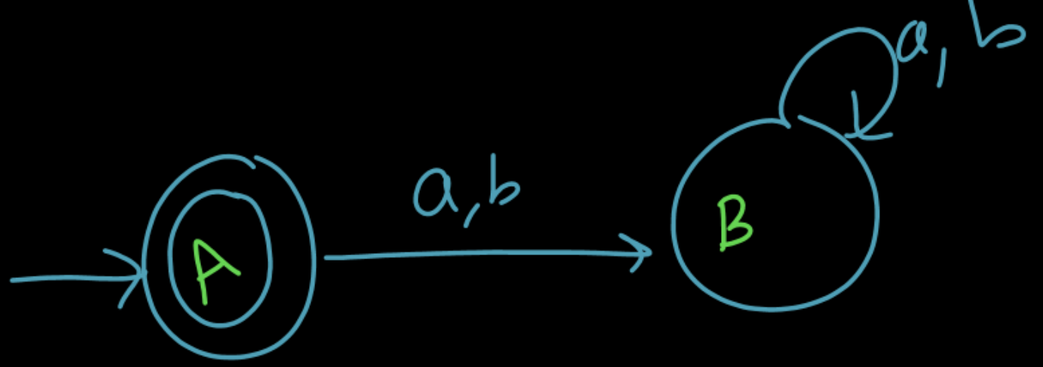




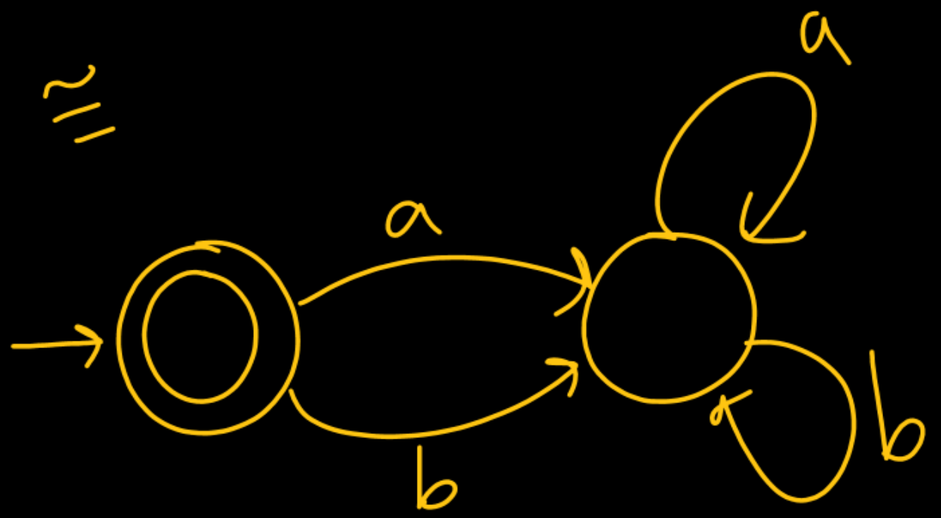








$\equiv$



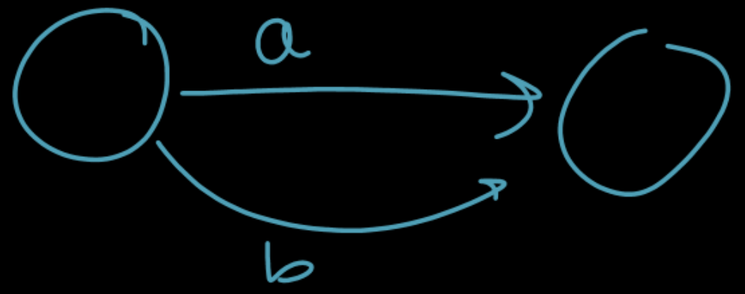
$\delta(A, a) = B$

$\delta(A, b) = B$

$\delta(B, a) = B$

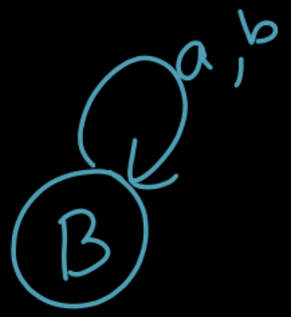
$\delta(B, b) = B$

Correct



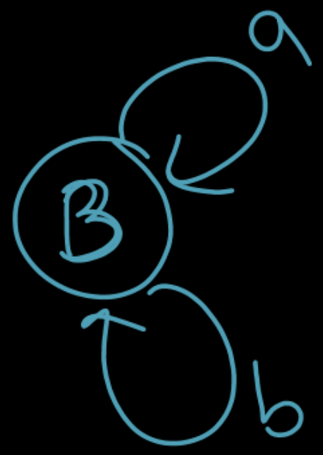
Never

1 edge



2 transitions

2 edges



2 transitions

① $|\delta| = |Q \times \Sigma|$

(δ: function, $Q \times \Sigma$: Domain)

② How many paths exist for every string in DFA ?

= 1 path

③ How many transitions in DFA ?

$= |\delta| = |Q \times \Sigma|$

✓ ε : A — one path

✗ a : A —a→ B — one path

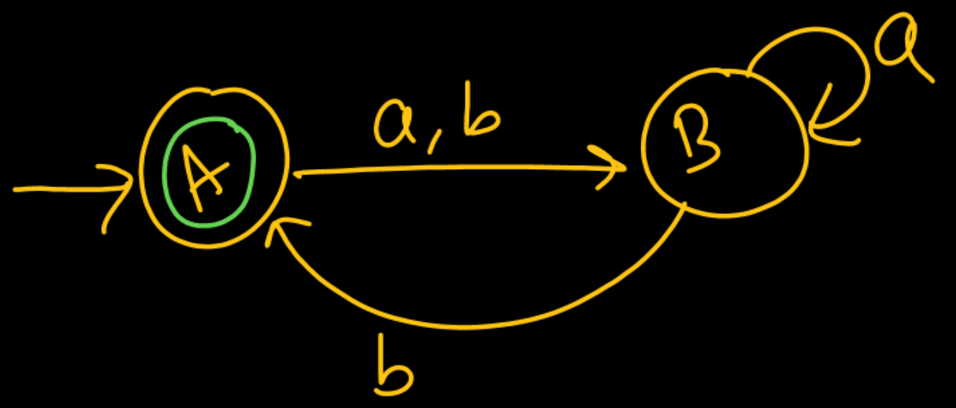
✗ b : A —b→ B — one path

✗ aa : A —a→ B —a→ B — one path

✓ ab : A —a→ B —b→ A — one path

✗ ba : A —b→ B —a→ B — one path

✓ bb : A —b→ B —b→ A



How many transitions ?

$= 4 = |\delta| = |Q \times \Sigma|$ (2 states, 2 symbols)

DFA with n states & K input symbols

$$\Downarrow$$

$$\text{No. of transitions} = nK$$

Important points:

I) If ε is valid string then initial state must be final

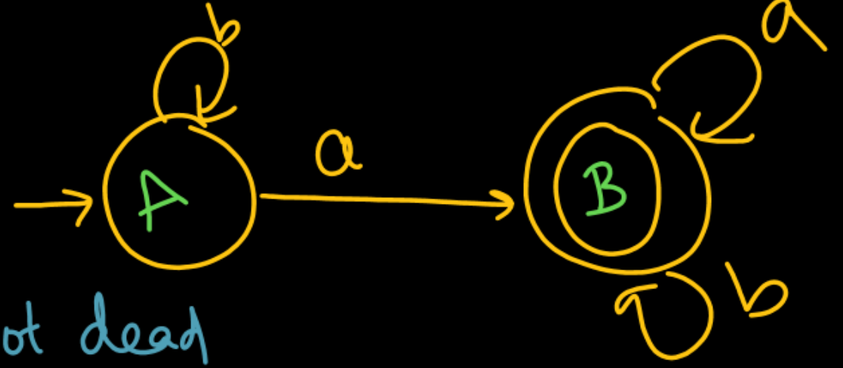
If initial state is final $\Rightarrow$ ε is accepted

If initial state is not final $\Rightarrow$ ε is not accepted

II) Dead state:

↳ It is non final but no path exist to the final state

A is non final

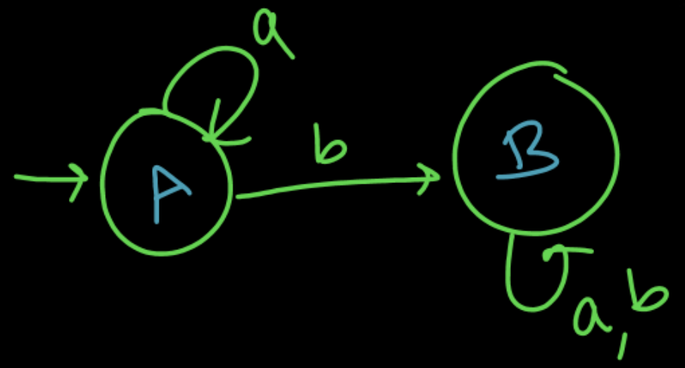


A is not dead

ε is not symbol

$A \overset{\varepsilon}{\Longrightarrow} A$: $\underbrace{A}_{\text{Path}}$

a: $\underbrace{A \xrightarrow{a} B}_{\text{Path}}$



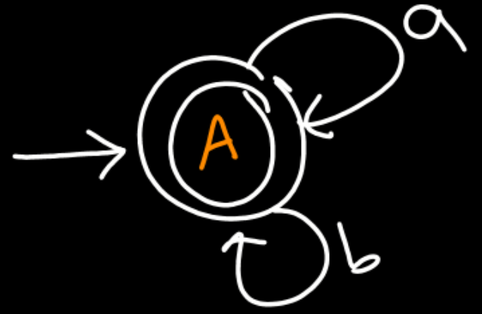
A is dead

B is dead

A and B are non final

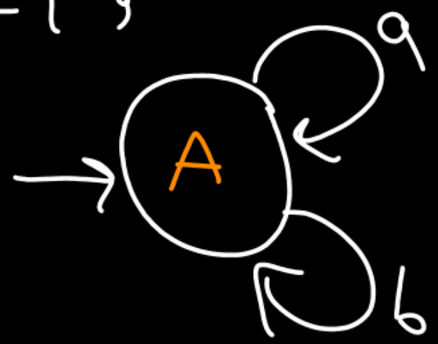
Mode-I : Special / Simple / easy

① $L = \Sigma^*$ over $\Sigma = \{a, b\}$



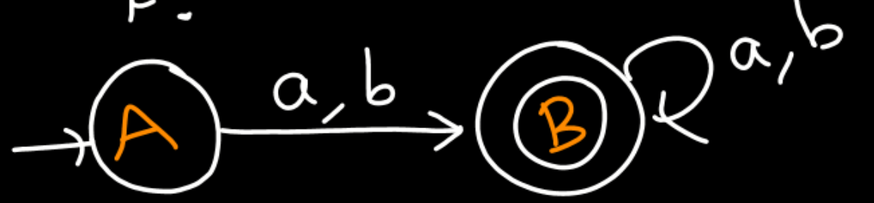
② $L = \phi$ over $\Sigma = \{a, b\}$

$= \{\}$

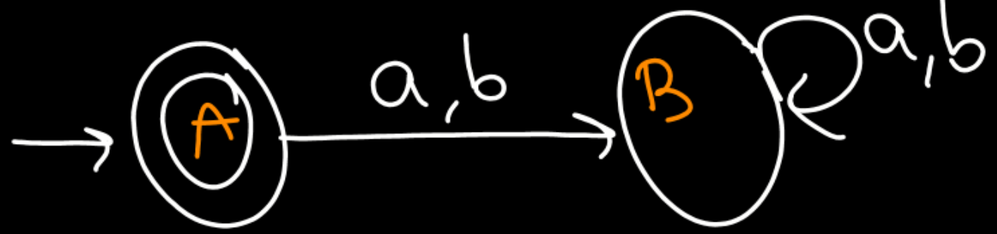


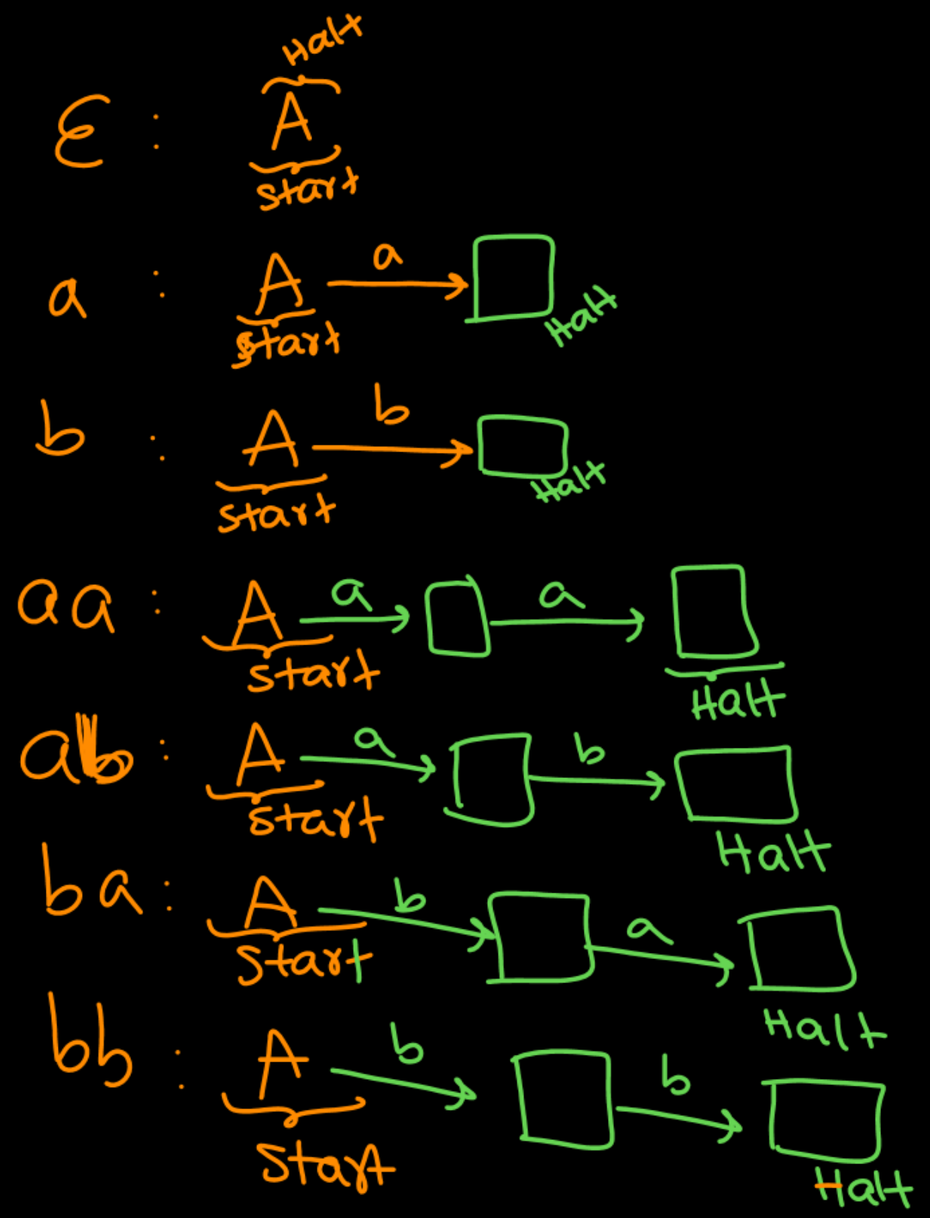
③ $L = \Sigma^+$ over $\Sigma = \{a, b\}$

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



④ $L = \{\varepsilon\}$ over $\Sigma = \{a, b\}$





If Halting state is final state

&

string is valid

(accepted)

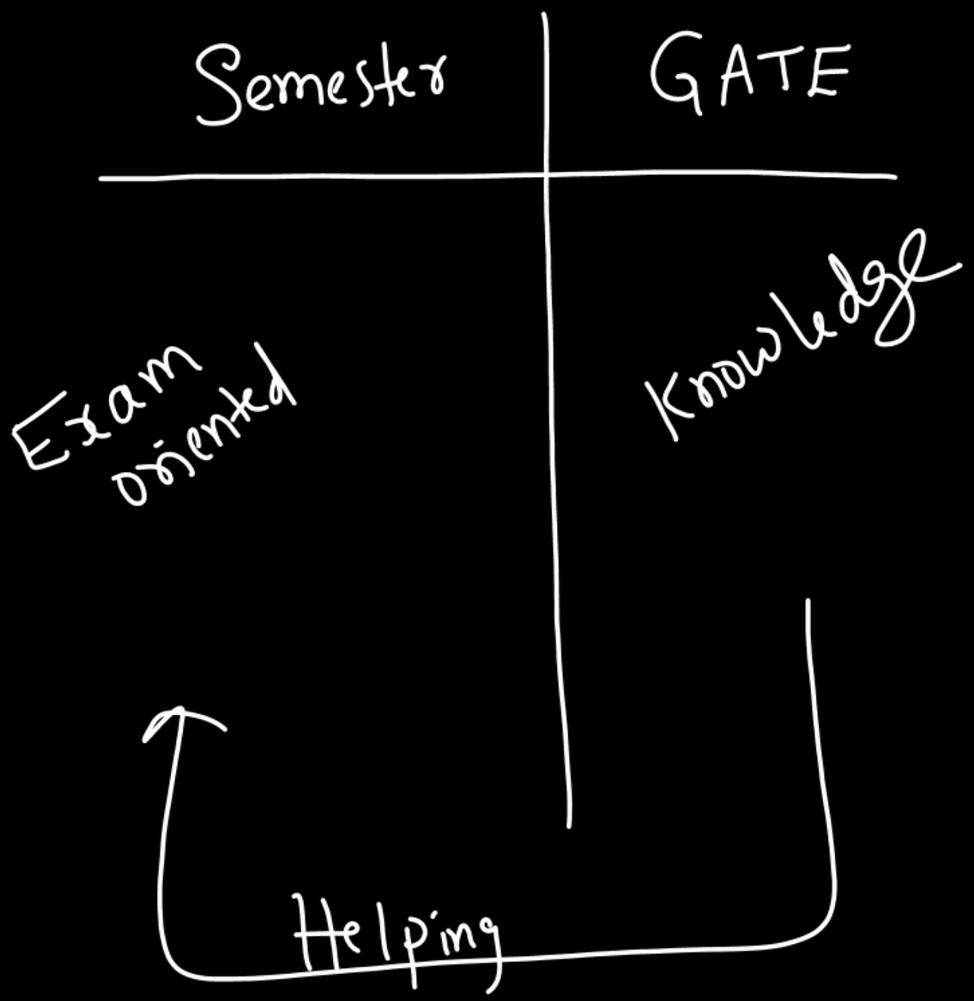
Summary

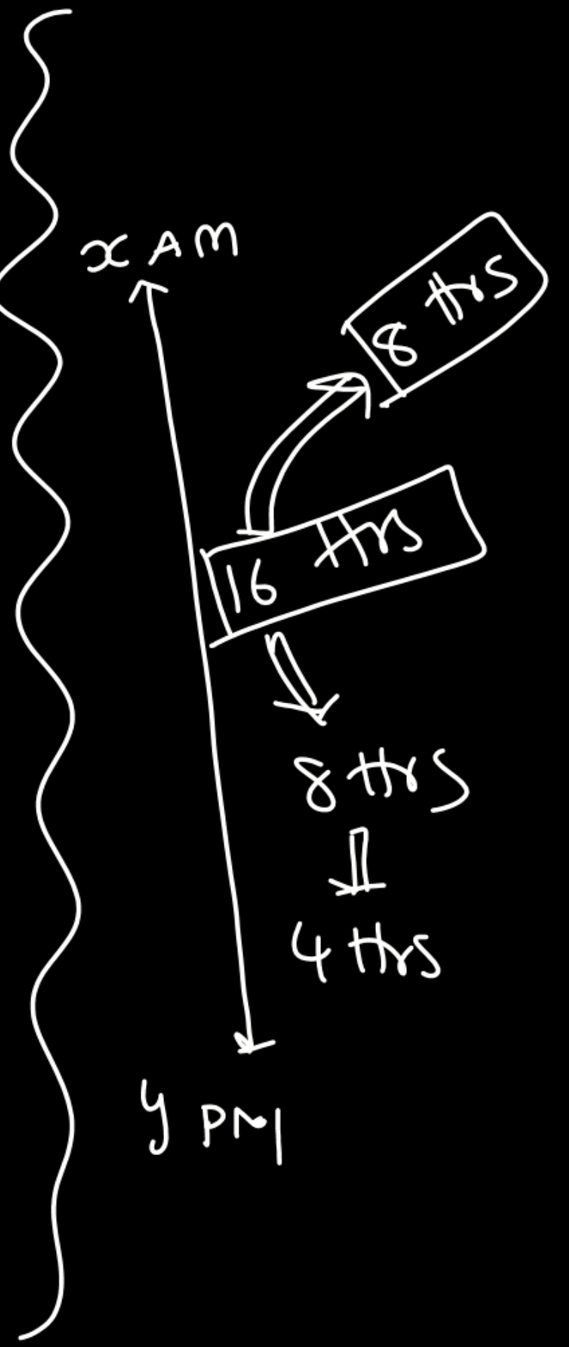
FA ✓

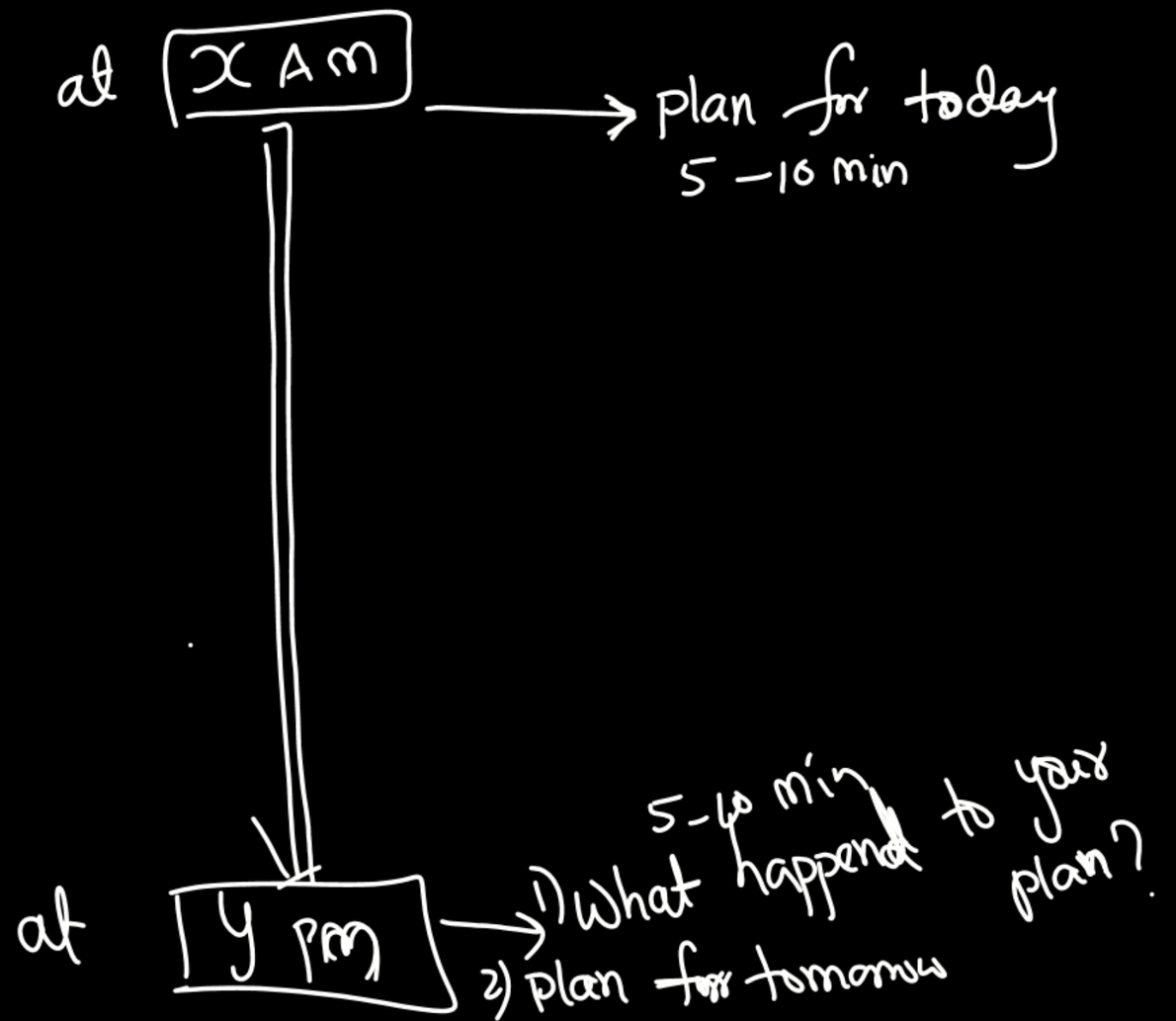
DFA ✓

⇓

How to construct DFA ?


Semester
GATE
Exam oriented
Knowledge
Helping


x AM
8 hrs
16 hrs
8 hrs
4 hrs
y PM


at
X AM
plan for today
5 – 10 min
at
Y PM
1) What happend to your plan?
5-10 min
2) plan for tomorrow

Thank you