CS & IT ENGINEERING

THEORY OF COMPUTATION

Regular Languages

Lecture No.- 04

By- Venkat sir

Recap of Previous Lecture

Topic Detection of Regular Languages

Topic Pumping Lemma

Topic

Topics to be Covered

Topic

Topic

Topic

{closure properties of Regular languages}

Topic : Pumping Lemma

To Prove a Language L is Non –Regular

Pumping length

1. Assume L is Regular
2. There exist F.A for L and n is number of states in that F.A
3. Select some string W from L such that |W|>=n .
4. Divide W into XYZ such that |xy|<=n and |y|>0.
5. Find a suitable integer i such that $\cancel{UV^iW}$ $xy^iz \notin L$ is not belongs to L.

3 & 5

Then L is not Regular.

$xy^iz \in L \quad \forall\, i \geq 0 \}$ then Regular

$L = |\{x_1, x_2 \cdots x_n\}| < K$

finite language

1 2 3 --- K

$X \geq K$

L

infinite language

closure Properties

Subset op

reg

$a^*b^* \subseteq (a+b)^*$

$\{a^n b^n\} \subseteq (a+b)^*$

NonReg

$\subseteq$ Regular language

{if always Regular then op is closed}

{Regular languages not closed under Subset op}

$$\{a^n b^n\} \subseteq \overset{\underline{\text{infinite}}}{\{a^n b^n c^m \mid n, m \geq 0\}}$$

(Q) Which of the following is true?

(a) Subset of regular set is always regular → $\boxed{\text{false}}$

$\square \subseteq \{a,b,c\}$

(b) " " any infinite set is always regular → false

(c) " " " finite set " " " → $\boxed{\text{true}}$

(d) none

finite $\subseteq \{a^n b^n \mid n \leq 5\}$

Regular

$L_1 \rightarrow \text{reg} \rightarrow r_1$

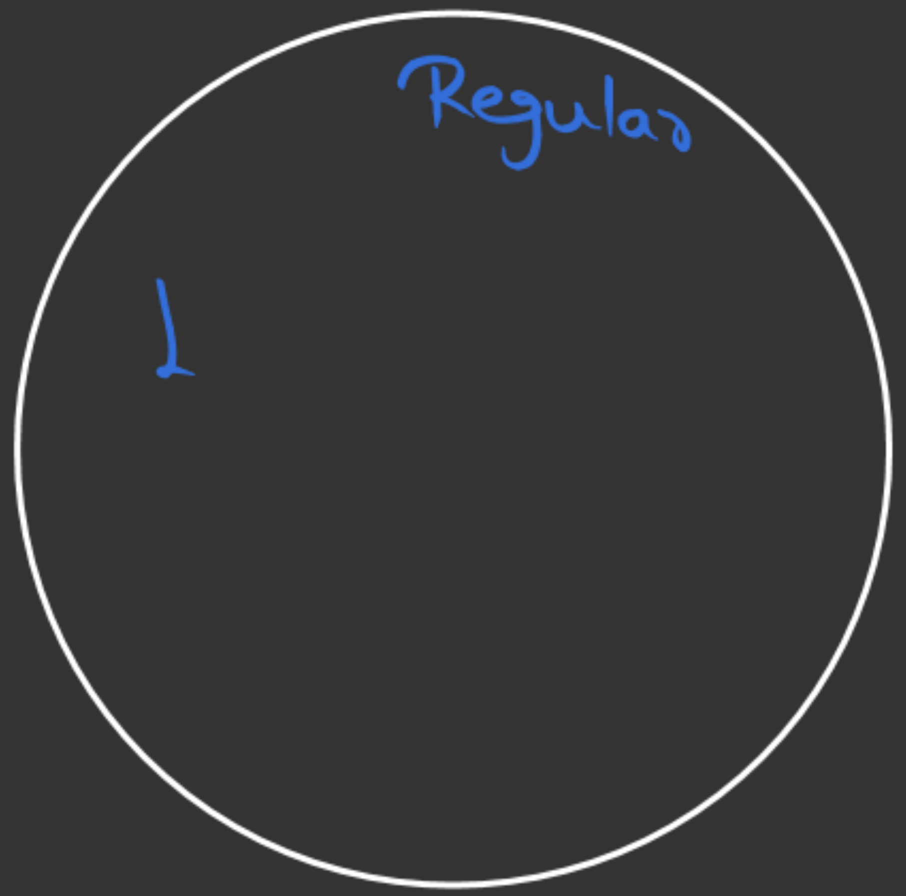
$L_2 \rightarrow \text{reg} \rightarrow r_2$

$L_1 \cup L_2 \Rightarrow$ always Regular

$\Downarrow$

$r_1 + r_2$

closed


Regular
L

$a^* b^*$

$L_1 = \{a^n b^m\}$

$L_2 = \{c^n d^m\}$

$c^* d^*$

$L_1 \cdot L_2$ → Reg

→ Non Reg

→ $a^* b^* c^* d^*$ ✓

Let $L_1 \rightarrow \{a^n\}_{n\ge 0} \rightarrow a^*$

$L_2 \rightarrow \{b^n\}_{n\ge 0} \rightarrow b^*$

Which of the following is false?

$L_1 \cdot L_2$ $\quad L_1 L_2 \neq a^n b^n$

$L_1 \cdot L = \{a^n b^m\}$

(a) $L_1 \cup L_2$ is Regular $\rightarrow$ true

(b) $L_1 \cdot L_2$ is Regular $\rightarrow$ true

$a^* b^*$

$\{a^n\}\{b^n\}$

$a^* \; b^*$

(c) L_1 is Regular $\rightarrow$ true

(d) none

$L_1 \rightarrow \text{reg} \rightarrow r_1$

$L_2 \rightarrow \text{reg} \rightarrow r_2$

$L_1 \cdot L_2 = r_1 \cdot r_2$

always Regular

$L \longrightarrow$ regular $\longrightarrow$ DFA

$\downarrow$

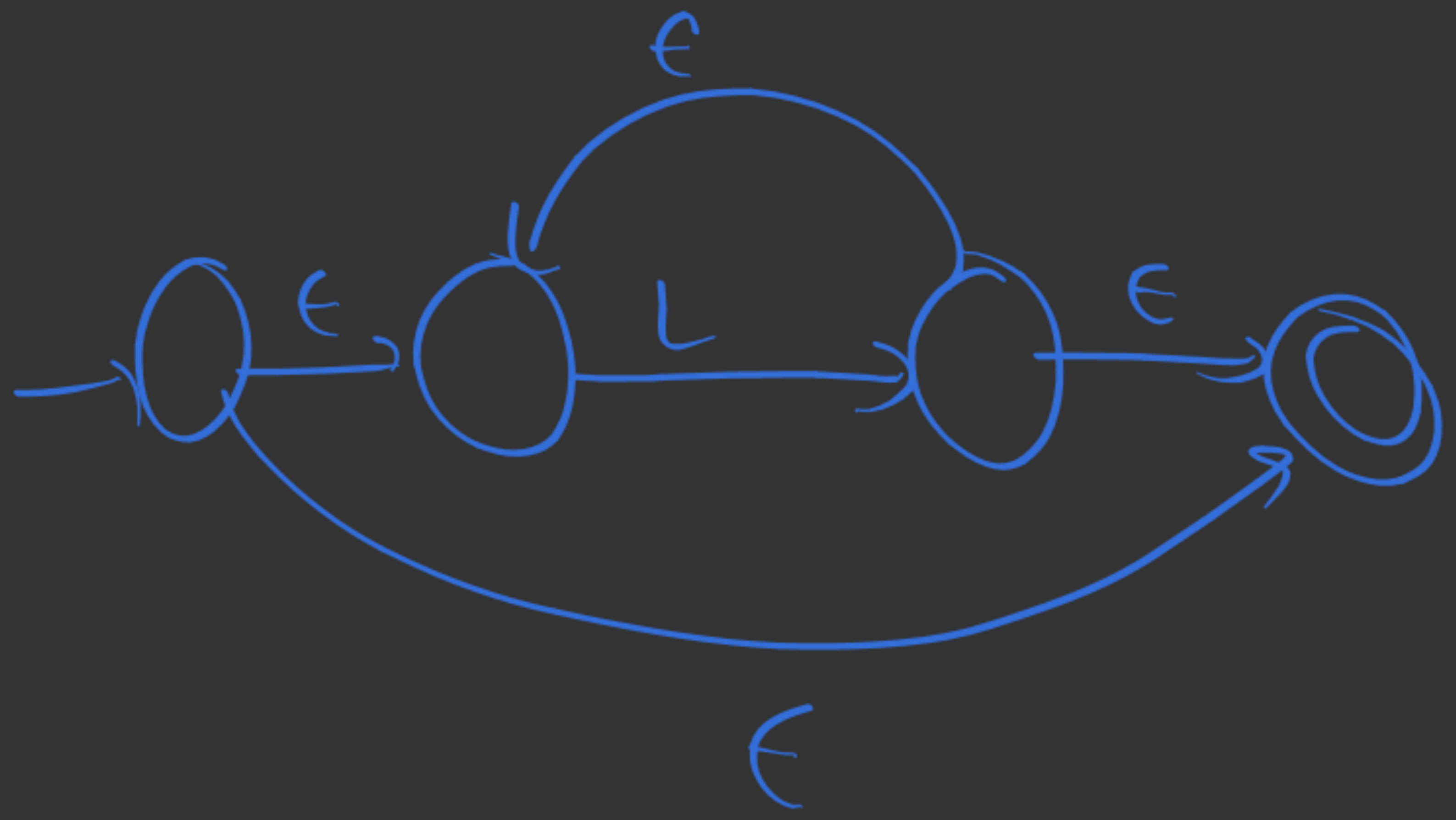
DFA^C

$\Sigma^* - L \longrightarrow$ DFA $\longleftarrow \text{DFA}^C$

regular

$L \to$ regular $\to$

$L^* \to$



(Q) How many states in $L_1 \cap L_2$ min DFA where

$L_1 = (a+b)^* a$

$L_2 = (a+b)^* b$

2 marks

(a) ~~1~~

(b) 2

(c) 4

(d) 6

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

$F_1 \times F_2$

Reversal op

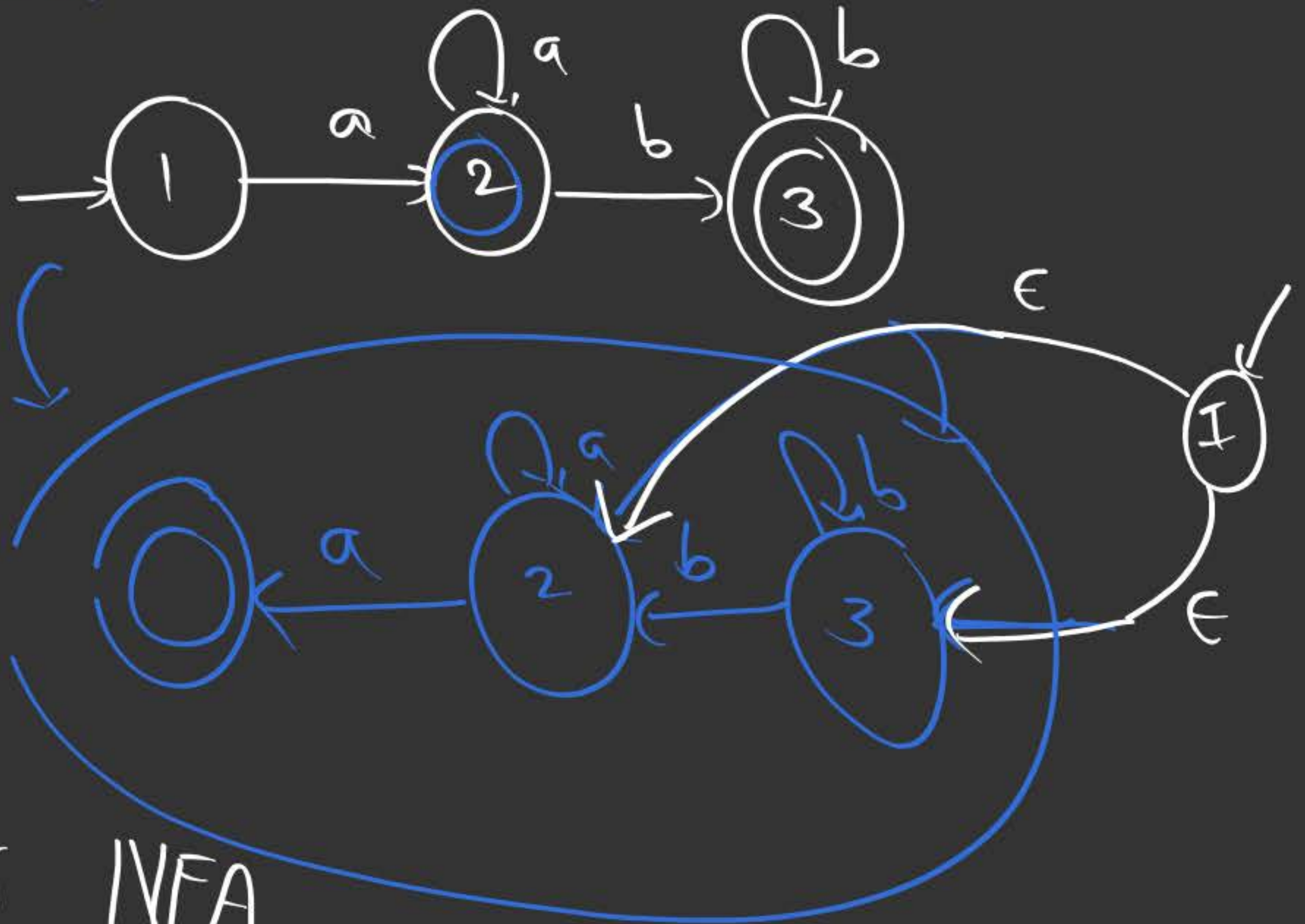
$L \to$ regular, $\{a^n b^m\}$

$L^R = $ DFA

DFA

L^R

$\{b^m a^n\}$



$L^R =$ always Regular

NFA

Suffix of a String : Sequence of trailing symbols over the given string.

$L \rightarrow R$

TOC
ϵ
C
OC
TOC

GATE
ϵ
E
TE
ATE
GATE

abc---n

=

Total no. of Suffixes

$(n+1)$

Prefix of a string :- Sequence of leading Symbols over the given string.

TOC
ϵ
T
TO
TOC

3 — 4

	GATE
1	ϵ
2	G
3	GA
4	GAT
5	GATE

abc - - - n

$\Downarrow$

Total no. of prefixes

$n+1$

	Regular language
1. Subset	✗
2. Union op	✓
3. Concatenation	✓
4. Complement	✓
5. Kleene closure	✓
6. Positive closure	✓

$L_1 \rightarrow \{a^n\}$

$L_2 \rightarrow \{b^n\}$

$L_1 \cdot L_2$

$\mathcal{L} = \{ab\}$

$L = ab$

<u>Suffixes</u> $- \{\epsilon, b, ab\}$

$$\text{Suffix}(L) = \text{Rev}(\text{prefix}(\text{Rev}(ab)))$$

$$= \text{Rev}(\text{prefix}(ba))$$

$$= \text{Rev}(\epsilon, b, ba)$$

$$= \epsilon, b, ab$$

(Q) How many states in $L_1 \wedge L_2$ min DFA

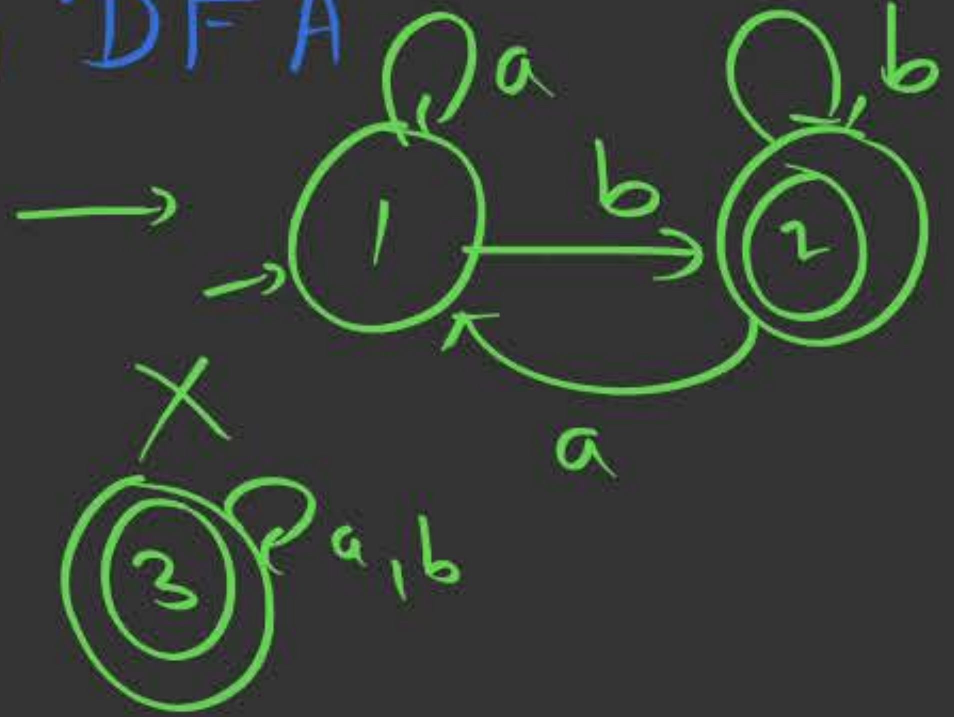
$L_1 \wedge L_2 = \boxed{(a+b)^* b}$

2

$L_1 = (a+b)^* b$ sub $\longrightarrow F_1 \longrightarrow$

$L_2 = (a+b)^*$ $\longrightarrow F_2$

super



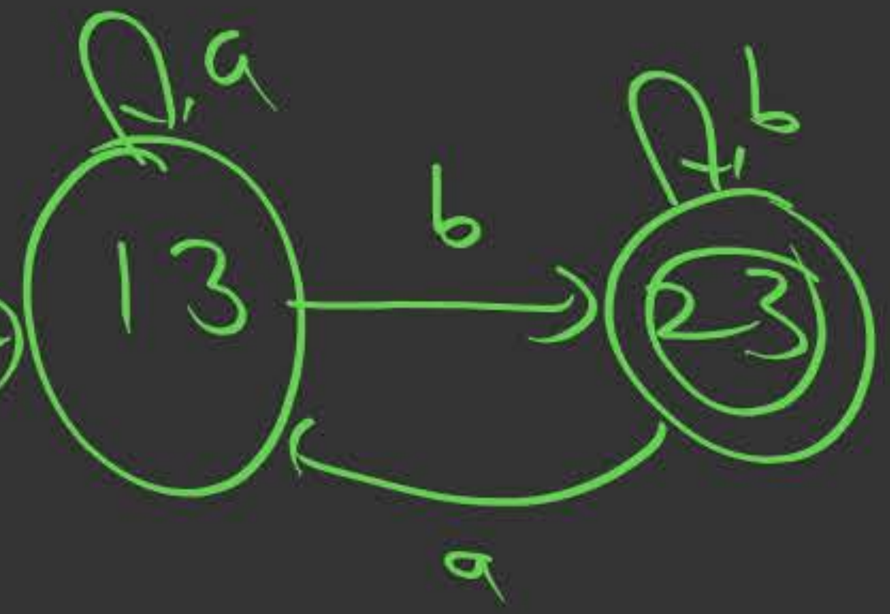
$L_1 \wedge L_2 = F_1 \times F_2$

$\Downarrow$

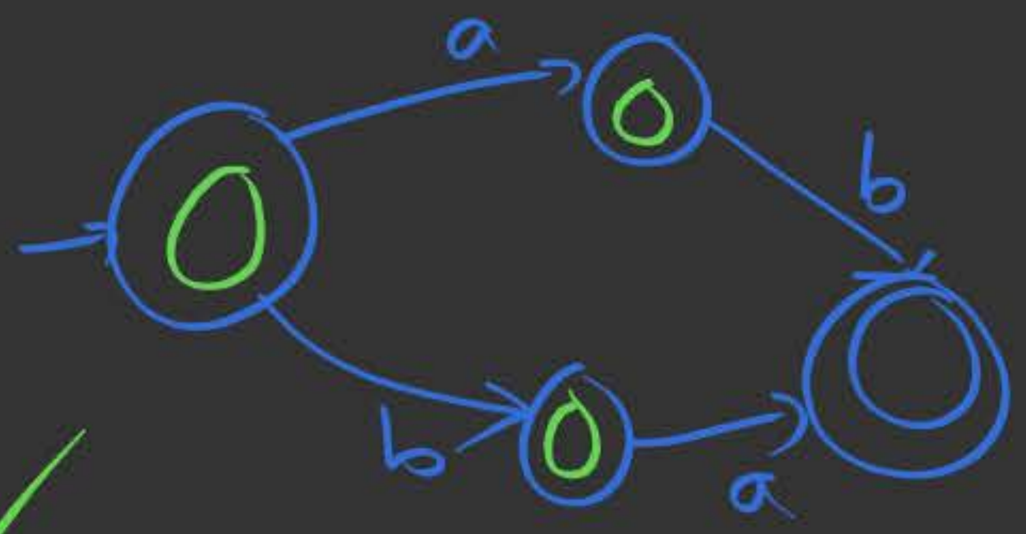
$F_1 \times F_2$

and

product FA



$L = \{ab, ba\}$



$\text{Prefix}(L) = \{\epsilon, a, ab, b, ba\}$

always Regular } Construct F.A by making all states as final states

	Regular Language	
7 Intersection op	✓	$(F_1 \times F_2)$
8 Difference op	✓	$\left(L_1 - L_2 = \underbrace{\overbrace{L_1 \wedge \overline{L_2}}^{\text{always reg}}}_{\text{reg} \wedge \text{reg}}\right)$
9 Reversal op	✓	(DFA Reversal can be Constructed)
10 prefix op	✓	(Construct DFA by making all states as final states)
11 Suffix op	✓	$\boxed{\text{Suffix}(L) = \text{reverse}(\text{prefix}(\text{reverse}(L)))}$

	Regular Language
12 Quotient op	✓
13 Substitution	✓
14 Homomorphism	✓
15 Inverse Homomorphism	✓

Homework

If $L_1 \wedge L_2$ is regular and L_1 is regular then L_2 is ___?___

$x+y=10$

$x=3$

$y=?$

(a) Always Regular

(b) need not be Regular

$L_1 \cup L_2 \rightarrow$ regular

regular $\cup$ reg

$(a+b)^*$ $\cup$ $a^n b^n \rightarrow (a+b)^*$

reg

reg

2 mins Summary

- **Topic** One
- **Topic** Two
- **Topic** Three
- **Topic** Four
- **Topic** Five

THANK - YOU