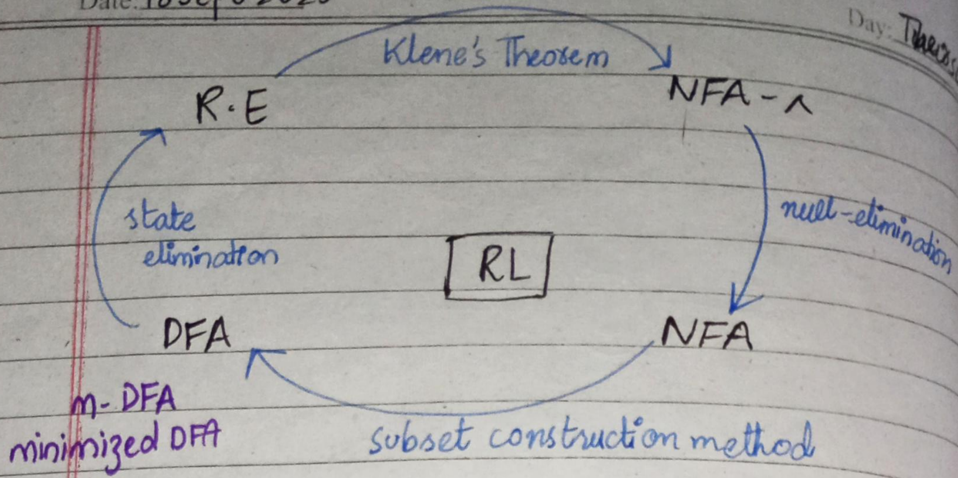
Lecture # 10

Date: 18 Sept 2025 Day: Thursd



Klene's Theorem:

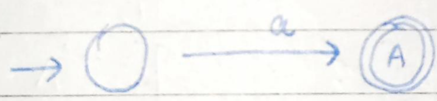
Consider the smallest language of the expression and make a machine for that.

$M3 = M_1 \cup M_2$

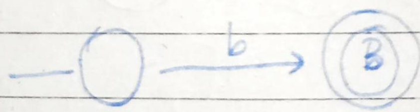
$$R.E = (a + b)^* \cdot b$$

a : M_1, b : M_2

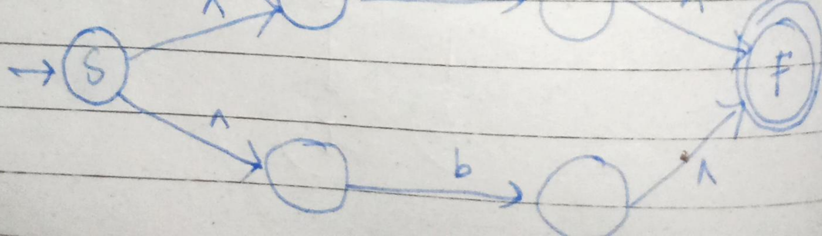
M_1 :



M_2 :

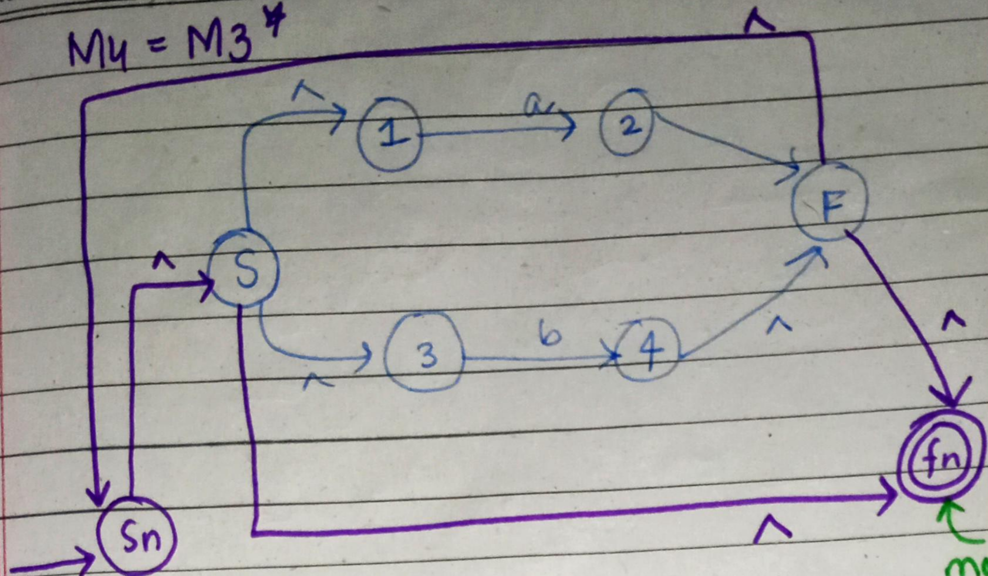


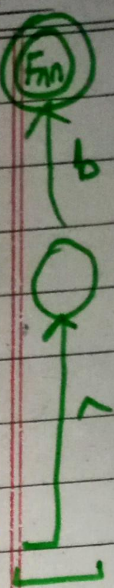
M_3 :



old S → new f...
old final → new start.

$M_4 = M_3^*$





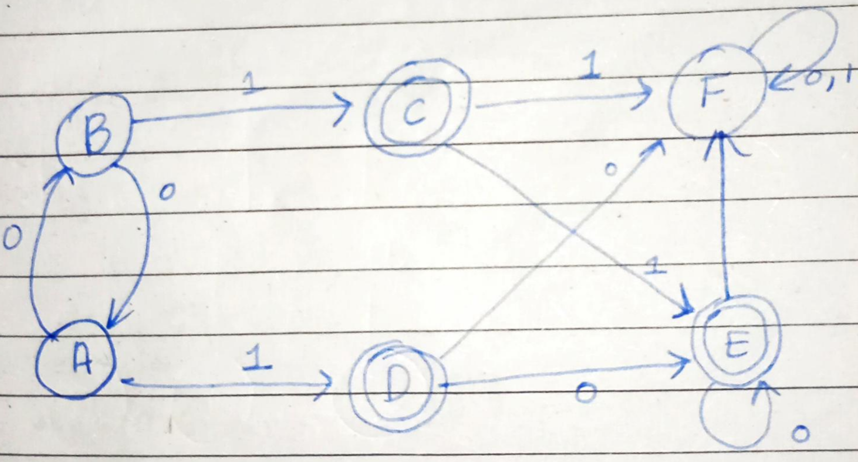
M5

make it non-final.

contatenation:

$M_5 = \bigcirc \xrightarrow{b} \circledcirc$ + star k liye old start sy new final ki transition remove.

$M_6 = M_4 \cdot M_5$



$0^* 1 0^*$

	A	B	C	D	E	F
A	X					
B	–	X				
C	1	1	X			
D	1	1		X		
E	1	1			X	
F	2	2	1	1	1	X

if both are from diff. sets:- . . . fill them

- Focus on one diagnol only

(1) Membership fn. fn'

Date: Day:

$A \to B$
$A \to D$

$D \xrightarrow{0} E$
$D \xrightarrow{1} F$

$B \to A$
$B \to C$

$E \xrightarrow{0} E$
$E \xrightarrow{1} F$

$C \xrightarrow{0} E$
$C \xrightarrow{1} F$

$F \xrightarrow{0} F$
$F \xrightarrow{1} F$

EE →
FF → } invalid...

$BA \xrightarrow{0} BA$
$BA \xrightarrow{1} DC$

source

Destination

$DC \xrightarrow{0} EE$
$DC \xrightarrow{1} FF$

if destination box is marked mark the source too.

$FA \xrightarrow{0} FB$
$FA \xrightarrow{1} FD$

FD → box was marked so marked the source FA ...

$FB \xrightarrow{0} FA$
$FB \to FC$

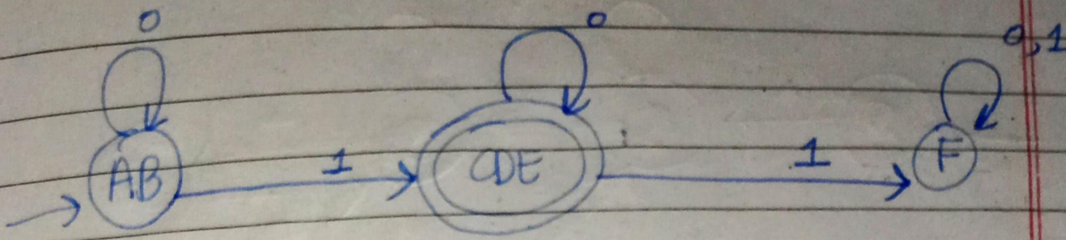
} min

will check for the next iteration (transitions); if the tables for any 2 iterations are same --- Stop

AB CDE F(left) ..

Date: Day:

Merge the empty bones states ...



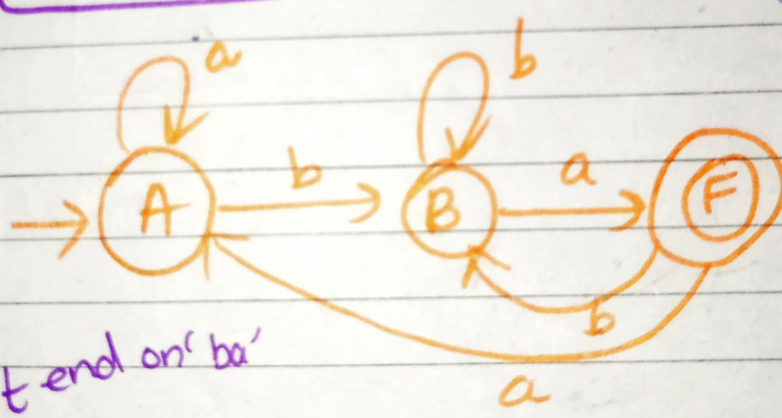
L = { x | x | ∈ Σ^* ; x doesn't ends on 'ba

→ ends on 'ba'

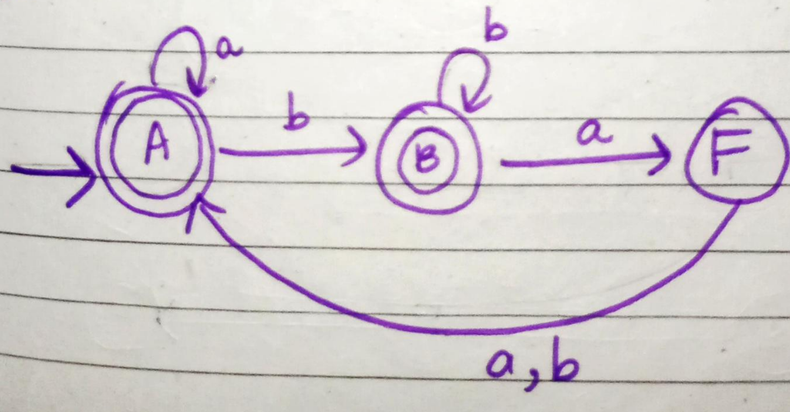
L = { ba, aba, bba, aaba, abba, bbba, baba, --- }

$\boxed{R.E = \Sigma^* ba}$

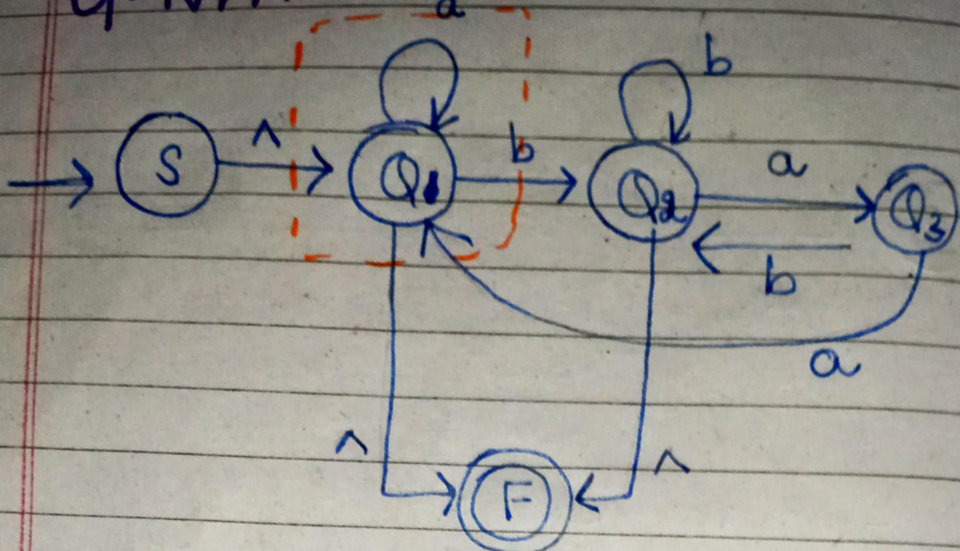
endswith "ba

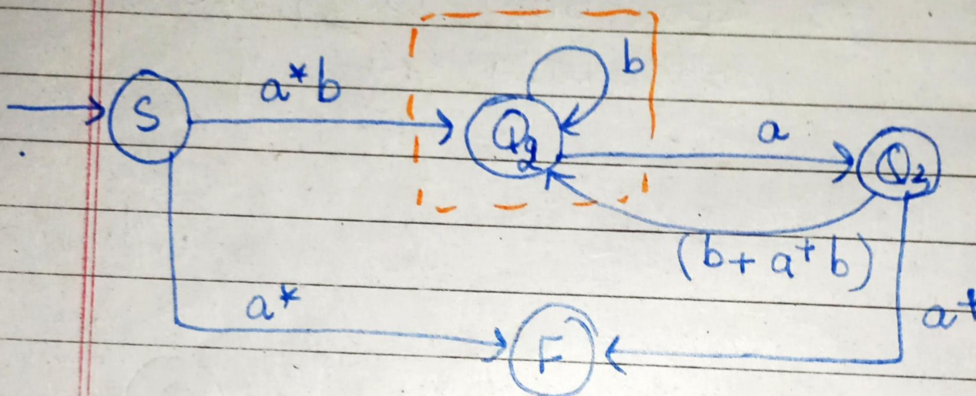


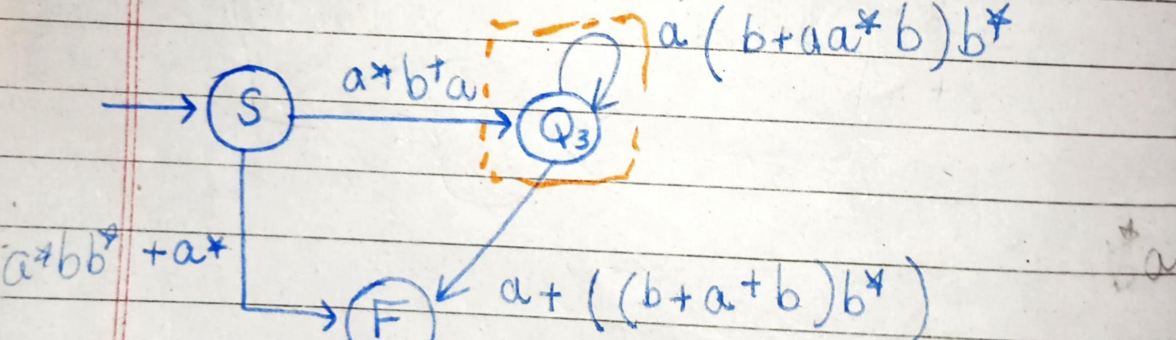
→ doesn't end on 'ba'

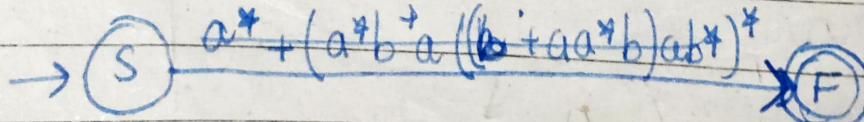


G-NFA:









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

Lecture #11

Date: Sep 25, 2025   Day: Thursday

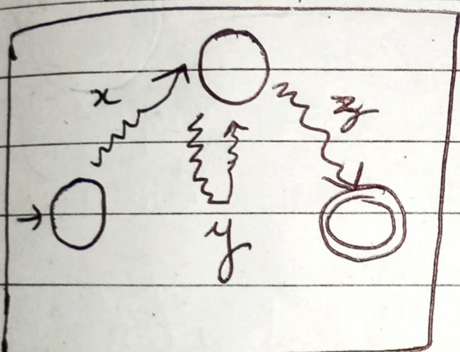
$L = \{ a^n b^m \; ; \; n > m \}$ Pumping Lamma for RL

$L = \{ a^n b^n \; ; \; n \geq 0 \}$ $\Sigma = \{a, b\}$

→ Requires memory
→ Not a regular language. (context free languages)

Pumping property (PP): All RL has pumping property.

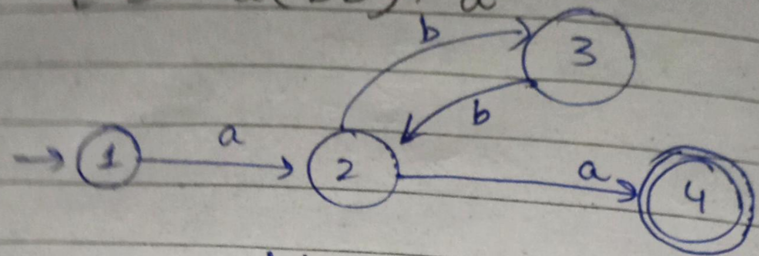
$|x| = \cancel{5} \simeq P \rightarrow$ 5 states



$x = w\vec{y}z$ Lopping element

1) $\forall x \in L \; ; \; |x| \geq P$ → no. of states
2) $\exists \; wyz$ such that $x = wyz$
3) $\forall \; i \geq 0 \quad wy^iz \in L$

$L = a(bb)^* a$



$x = abba$ $\quad |x| = 4 \Rightarrow P$

③ $|wy| \leq p$

$w = a$

$y = b$

$z = a$

② $|y| \geq 1$

Looping element can't be null

$a(bb)^i a$

if $i = 0$ 'aa' $\notin L$ $\Rightarrow$ pumping down.

if $i = 1$ 'abba' $\Rightarrow$ No pumping effect

if $i = 2$ 'abbbba' $\Rightarrow$ Pumping up

☆ Pumping Lemma is always used to prove a "not a regular language."

→ not to prove a regular language

prefix

$w = ab$

$y = bb$

$z = ba$

only need one counter example to prove that it is not a RL

Date:

Day:

$\sim PP \longrightarrow \sim RL$

1) $\exists x \in L \; ; \; |x| \geq P$
2) $\forall wyz$ such that $x = wyz$
3) $\exists \; i \geq 0$ when $wy^i z \in L$

so $L \in RL$

$L = \{ a^n b^n \; ; \; n \geq 0 \}$

$x = a^P b^P$ $\quad |x| = 2P$

Only one variation

$w = a^{P-1}$

$y = a$

$z = b^P$

$\rightarrow a^{P-1} (a)^i \; b^P$

so $L \in RL$.

if $i = 0 \Rightarrow a^{P-1} b^P \notin L \Rightarrow$ pump down

↳ disproved.

for all variations

$w = a^{P-s}$ $\quad P \geq s \geq 1$

$y = a^s$

$z = b^P$

1) select a string

2)

$a^{P-s} (a^s)^i \; b^P$

$i = 0$ $\quad$ As $s \geq 1$ so pick $s = 1$