Regular grammar to regular expression
Regular expression to regular grammar
Regular language to regular grammar
Regular grammar to language

Regular expression to finite automata
Adren's theorem (finite automata to regular expression)

Regular Grammar to Regular expression

1) S -> aSB   L = { a^n b^n | n>=2 }
   S -> aB
   B -> b

2) S-> aS | epsilon   L ={ a^n | n>0 }

Regular Expression to regular grammar

1) (a+b)*a
S → WX
W → aW | bW | epsilon
X → a

2) (a+1)*
S → aS | 1s | epsilon

3) a*
S → aS | epsilon

4) a+0
S → a|0

5) (ab)*
S → abS | epsilon

6) (a/b)*
S → aS|bS|epsilon

7) (a*) + (1)+ (b*)

S → aS|1|bS|epsilon

8) (a*) + (1) + (b+)
S → aS|1|b|epsilon

9) (a/b) (a/b) (a/b)*

S → XXY

X → a|b

Y → aY | bY | epsilon

10) (a/b/epsilon) (a/b/epsilon)

S → XX

X → a|b|epsilon

11) a (a/b)* b

S → aXb

X → aX | bX | epsilon

12) L = {a^n b^n} | n>=0

S → aSb | epsilon

13) L = {a^n b^m} | n,m>=0

S → aAbB

A → aA|epsilon

B → bB | epsilon

13) L = { (ab)^n } | n>=0

S → abS

14) L = { a^n b^n c ^m | n,m>=0 }

S → aXbY | epsilon

X → aXb | epsilon

Y → cY | epsilon

15) L = { a^n b^n c ^m | n,m>=1 }

S → aXbcY

X → aXb | epsilon

Y → cY | epsilon

16) L = { a^n c ^m b^n | n,m>=0 }

S → aSb | aXb |X | epsilon

X → cX | epsilon

16) L = { a^n c ^m b^n | n,m>=1 }

S → aSb | aXb

X → cX | c

17) L = { a^n c ^m b^n | n,m>=1 }

S → aSb | aXb

X → cX

E = {a,b}

L = { all non empty string starts and ends with the same symbol }

S → aXa | bXb | a | b

X → aX | bX

E = {a,b}

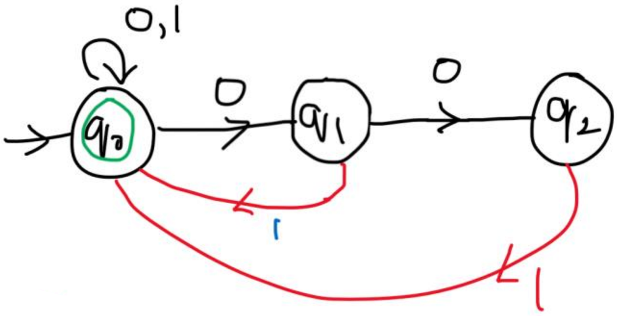
L = { Palindrome }

S → aXa | bXb | a | b | epsilon

X → aX | bX | epsilon

Regular language to Regular grammar

1)L={w | w ends with 00}



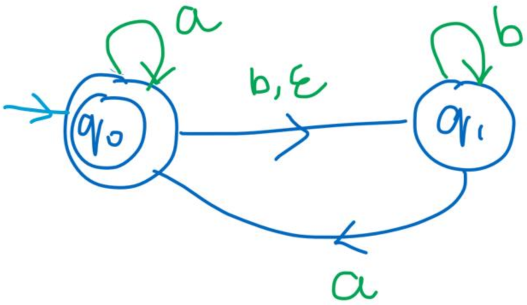
$S \rightarrow 0A \mid 1A$

$A \rightarrow 0A$

$B \rightarrow 0$

2)L={0*,1*,0*} with 3 states

3)L={a^n union b^na | n>=0}



$S \rightarrow aA \mid B \mid \varepsilon$

$A \rightarrow aA \mid \varepsilon$

$B \rightarrow bB \mid a$

*** L={ (a^n | n>=1) union (b^ma^k | m,k>=0) }**

*** L(G) = { a^nca^n / n ≥ 0 }**

*** L(G) = { a^nb^{2n} / n ≥ 0 }**

*** L(G) = { $a^{n+2}b^n$ / n ≥ 1 }**

*** L(G) = { a^nb^{n-3} / n ≥ 3 }**

*** L(G) = { a^nb^m / n ≥ 0, m > n }**

The grammar can gone any way, but wrong string should not pass through that grammar.

$S \to aXbY \mid bY$

$X \to aXb \mid \varepsilon$

$Y \to bY \mid \varepsilon$

*** L(G) = { $a^n b^n c^m$ / n, m ≥ 1 }**

$S \to XY$

$X \to aXb \mid ab$

$Y \to cY \mid c$

*** L(G) = { $a^n b^n c^m d^m$ / n, m ≥ 1 }**

$S \to XY$

$X \to aXb \mid ab$

$Y \to cYd \mid cd$

*** L(G) = { $a^n b^m c^m d^n$ / n, m ≥ 1 }**

*** L(G) = { $a^n b^m$ / n, m ≥ 1 m ≠ n}**

*** L(G) = { w {a, b}* / $n_a(w) = n_b(w) + 1$ }**

1)L = { a^n b^m | n,m>=1 }

2) **L = { a^n b^n c^m | n,m>=1 }**

S → XY

X → aXb | ab
Y → cY | c

3) **L = { a^n c^m b^n | n,m>=1 }**
S → aXb
X → aXb | CX | c

4) **L = { a^n b^m a^2n | n,m>=0 }**
S → aXaa |epsilon
X → aXaa |bX | epsilon

5) **E = {a,b}**
All non empty strings start and ends with the same symbol (or) Palindrome
S → aAa | bAb | a |b | epsilon
A → aA | bA | epsilon

6) **L = { w ∈ (0,1)* | w^R and |w| is even }**
S → epsilon | 1S1 | 0S0

7) **L = { w ∈ (0,1)* | w contains at-least 3 ones }**
S → A1|A1|A1
A → epsilon | 0A | 1A

8) **L = { $a^i b^j c^k$ | i,j,k>=0 and i+j = k }**
S → aSc | X
X → bXc | epsilon

9) **L = { $a^i b^j c^k$ | i,j,k>=0 and i=j }**
S → aXbcC
X → aXb | epsilon
C → cC | epsilon

10) $L = \{ a^i b^j c^k \mid i,j,k>=0 \text{ and } i=k \}$

S → aXc

X → bX | aXc | epsilon

11) $L = \{ a^i b^j c^k \mid i<j \}$

12) $L = \{ a^i b^j c^k \mid i<k \}$

Regular grammar to Regular language

1)

S → aS (rule 1)

S → a (rule 2)

$S \Rightarrow a$

$S \Rightarrow aS \Rightarrow aa$

$S \Rightarrow aS \Rightarrow aaS \Rightarrow aaa$

2)

S → aS (rule 1)

S → b (rule 2)

$S \Rightarrow b$

$S \Rightarrow aS \Rightarrow ab$

$S \Rightarrow aS \Rightarrow aaS \Rightarrow aab$

3)

S → aSb (rule 1)

S → ab (rule 2)

$S \Rightarrow ab$

$S \Rightarrow aSb \Rightarrow aabb$

$S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aaabbb$

4)

S → aSa (rule 1)

S → bSb (rule 2)

S → c (rule 3)

$S \Rightarrow c$

$S \Rightarrow aSa \Rightarrow aca$

$S \Rightarrow bSb \Rightarrow bcb$

$S \Rightarrow aSa \Rightarrow abSba \Rightarrow abcba$

$S \Rightarrow bSb \Rightarrow baSab \Rightarrow bacab$

5)

$S \rightarrow aSBc$ (rule 1)

$S \rightarrow abc$ (rule 2)

$cB \rightarrow Bc$ (rule 3)

$bB \rightarrow bb$ (rule 4)

$S \Rightarrow \mathbf{abc}$

$S \Rightarrow aSBc \Rightarrow aabcBc \Rightarrow aabBcc \Rightarrow \mathbf{aabbcc}$

$S \Rightarrow aSBc \Rightarrow aaSBcBc \Rightarrow aaabcBcBc$

$\Rightarrow aaabBccBc \Rightarrow aaabBcBcc \Rightarrow aaabBBccc$

$\Rightarrow aaabbBccc \Rightarrow \mathbf{aaabbbccc}$

6)

$S \rightarrow AB$ (rule 1)

$A \rightarrow aAb$ (rule 2)

$bB \rightarrow bbbB$ (rule 3)

$aAb \rightarrow aa$ (rule 4)

$B \rightarrow \varepsilon$ (rule 5)

$S \Rightarrow AB \Rightarrow aAbB \Rightarrow aaB \Rightarrow aa\varepsilon = \mathbf{aa}$

$S \Rightarrow AB \Rightarrow aAbB \Rightarrow aAbbbB \Rightarrow aAbbb\varepsilon$

$\Rightarrow \mathbf{aabb}$

$S \Rightarrow AB \Rightarrow aAbB \Rightarrow aAbbbB \Rightarrow aAbbb\varepsilon$

$\Rightarrow aaAbbbb$

$\Rightarrow \mathbf{aaabbb}$

7)

$S \rightarrow AB$ (rule 1)

$A \rightarrow aAb$ (rule 2)

$bB \rightarrow bbbB$ (rule 3)

$aAb \rightarrow aa$ (rule 4)

$B \rightarrow \varepsilon$ (rule 5)

$S \Rightarrow AB \Rightarrow aAbB \Rightarrow aaB \Rightarrow aa\varepsilon = \mathbf{aa}$

$S \Rightarrow AB \Rightarrow aAbB \Rightarrow aAbbbB \Rightarrow aAbbb\varepsilon$

$\Rightarrow \mathbf{aabb}$

$S \Rightarrow AB \Rightarrow aAbB \Rightarrow aAbbbB \Rightarrow aAbbb\varepsilon$

$\Rightarrow aaAbbbb$

$\Rightarrow \mathbf{aaabbb}$

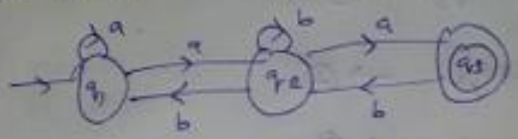
$L(G) = \{ a^{n+1}b^{n+k} / n \geq 1, k = -1, 1, 3, 5, \ldots\}$

1)

Arden's theorem:

$R = Q + RP$ $\boxed{R = QP^*}$

Convert a FA to regular expression



$q_3 = q_2 a \longrightarrow (1)$

$q_2 = q_1 a + q_2 b + q_3 b \longrightarrow (2)$

$\rightarrow q_1 = \varepsilon + q_1 a + q_2 b \longrightarrow (3)$

$q_2 = q_1 a + q_2 b + q_3 b$

$= q_1 a + q_2 b + (q_2 a) b$

$q_2 = q_1 a + q_2 (b + ab)$

$R \quad Q \quad + \quad RP$

$q_2 = q_1 a (b + ab)^*$

$q_1 = \varepsilon + q_1 a + q_2 b$

$= \varepsilon + q_1 a + (q_1 a (b + ab)^*) b$

$= \varepsilon + q_1 (a + (a (b + ab)^* b))$

$q_1 = \varepsilon (a + (a (b + ab)^*) b))$

$q_1 = \varepsilon [a + a (b + ab)^* b]^*$

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

$q_3 = q_2 a$

$= q_1 a (b + ab)^*$

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

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

2)

$$q_3 = q_2 0 + q_3 0 + q_3 1 \rightarrow (1)$$

$$q_2 = q_1 1 + q_2 1 \rightarrow (2)$$

$$q_1 = \varepsilon + q_1 0 \rightarrow (3)$$

$$\boxed{\begin{aligned} q_3 &= q_2 0 + (q_3 (0+1)) \\ &= q_2 0 (0+1)^* \end{aligned}}$$

q_3 is trap / unreachable state

$$q_1 = \varepsilon + q_1 0$$

$$q_1 = \varepsilon 0^*$$

$$q_1 = 0^*$$

$$q_2 = q_1 1 + q_2 1$$

$$q_2 = q_1 1 1^*$$

$$= q_1 1 1^*$$

$$\boxed{q_2 = 0^* 1 1^*}$$

When, we have more than one final state, we have to take the union

$$\boxed{\therefore 0^* + 0^* 1 1^*}$$

$0^*(\varepsilon + 11^*)$

$0^*(1^*)$

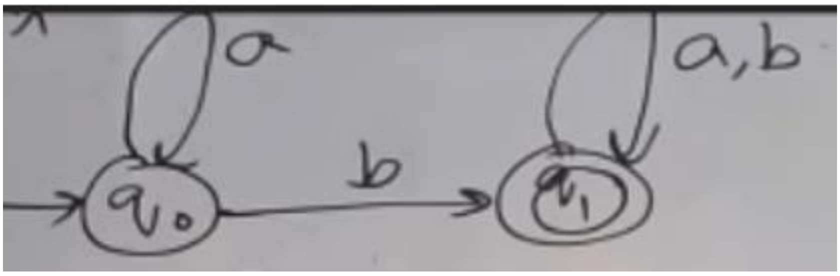
$$0^* (\varepsilon + 11^*)$$

$$\boxed{\Rightarrow 0^* 1^*}$$

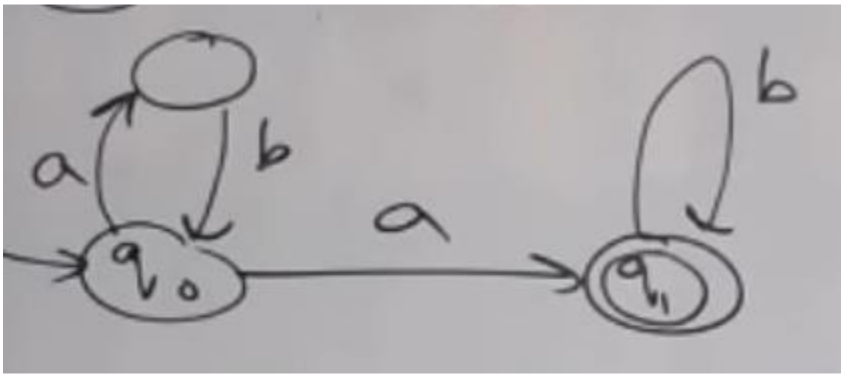
Regular expression to finite automata

1) 0*10*   L={ w|w contains a single 1 }
2) (0|1)* 1 (0|1)* L = { w|w contains at-least a single 1 }
3) (01) U (10)
4) (0|1 0|1 0|1)*
5) 0(0|1)*0 + 1(0|1)*1   + 0 + 1
6) Null symbol
7) Epsilon symbol
8) a+b
9) a.b
10) a*
11) a+
12) (a + b)*
13) a*b*
14) (ab)*
15) a*b
16) ab*
17) a*bc*

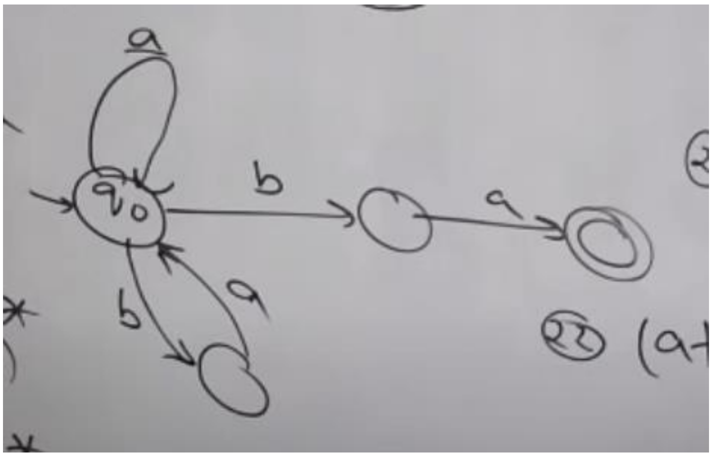
18) a*b(a+b)*



19) (ab)*ab*

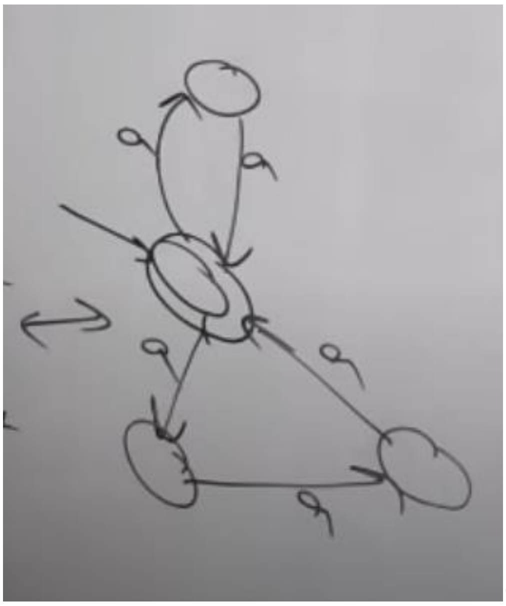


20) (a+ba)*ba

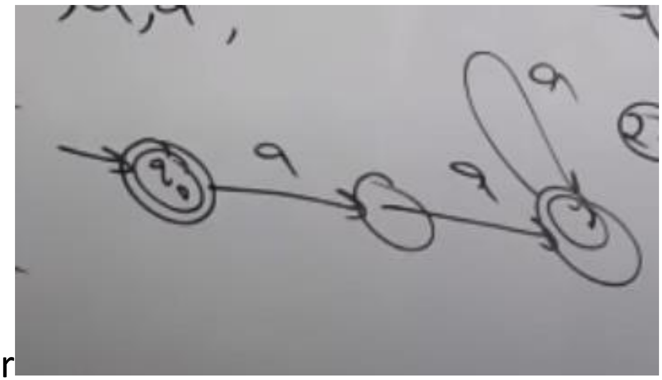


21) (aa+aaa)*

ε, $\cancel{A^1}$, A^2, A^3, A^4

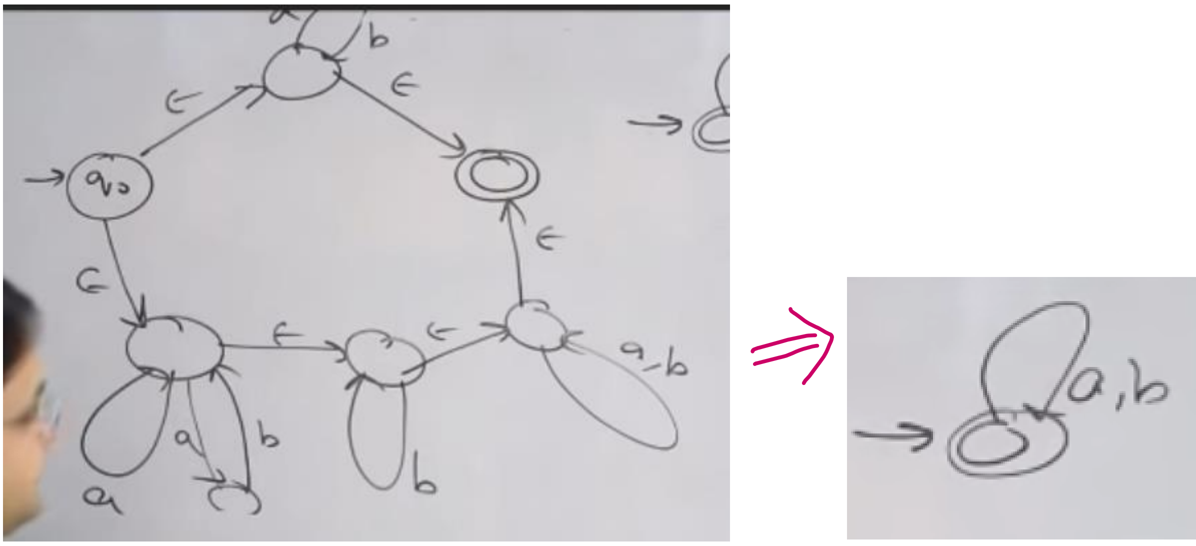


(or

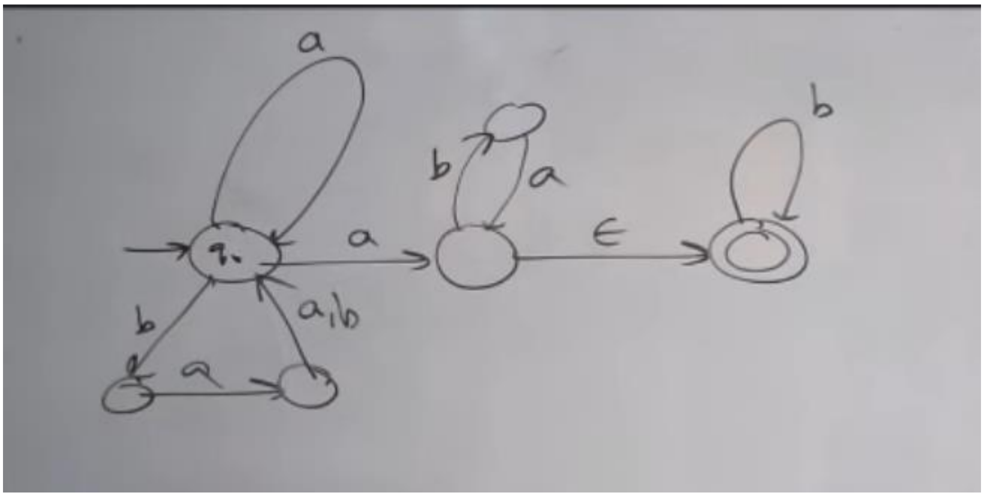


22) (a+aaaa)*

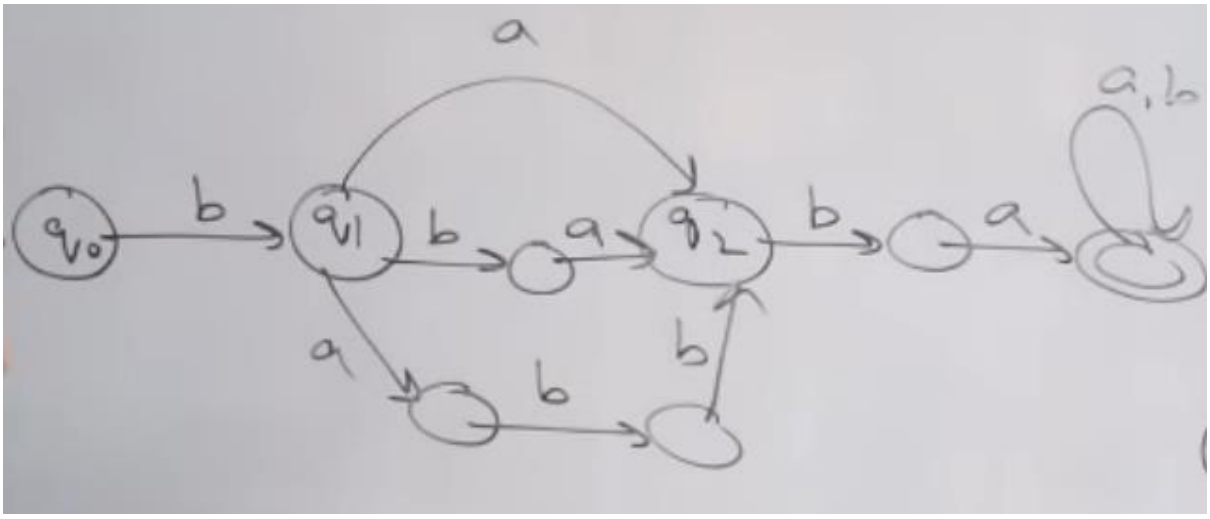
21) (ab)* + (a+ab)*b*(a+b)*



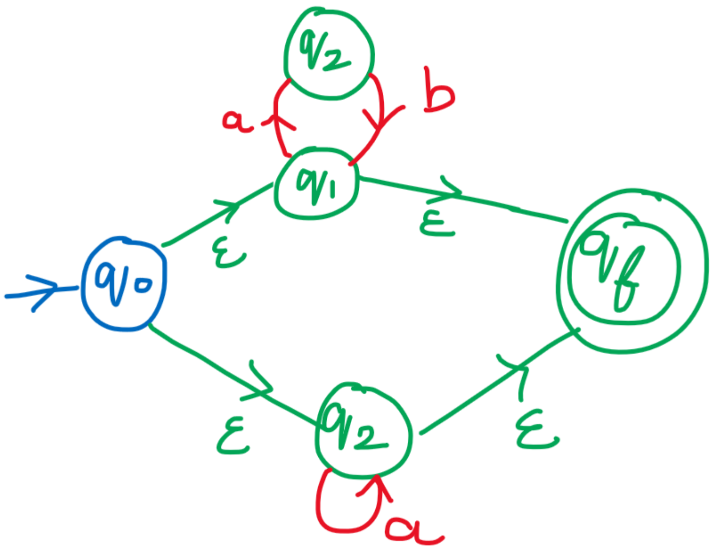
22) [a+ba(a+b)]* a(ba)* b*



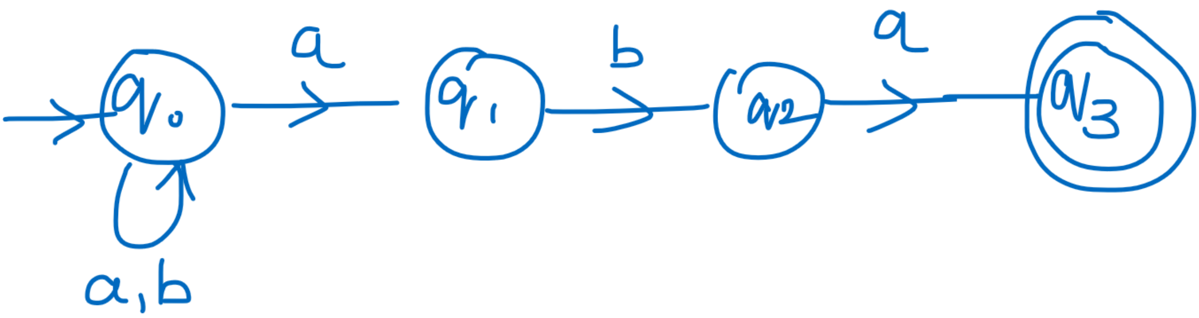
23) b(a+ba+abb) (ba(a+b)*)



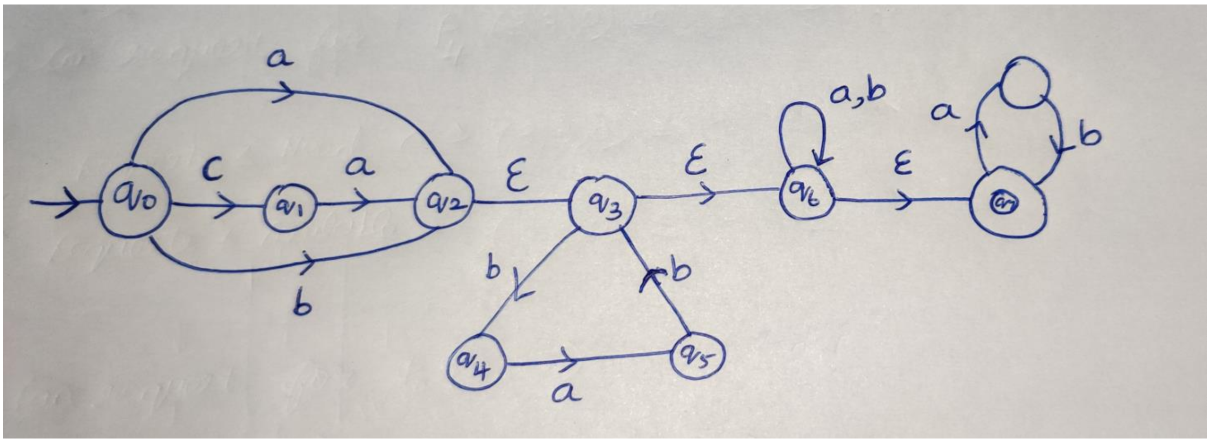
24) (ab U a)*



25) (a U b)* aba



26) (a+b+ca) [(bab)* + (a+b)*]* (ab)*



If the input alphabet is ∑ then the regular expression

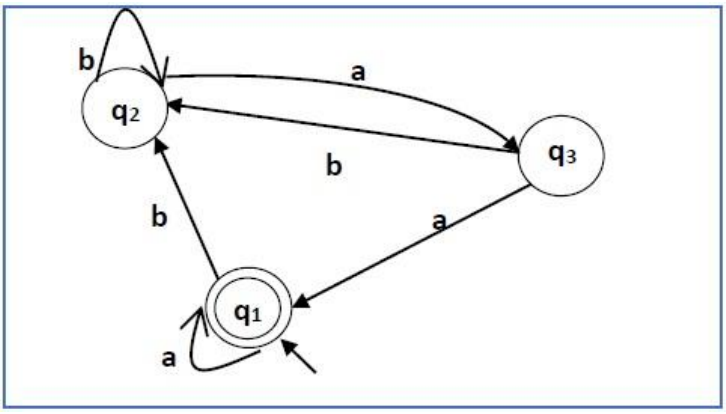
∑ → describes the language consisting of all strings of length 1 over this alphabets

∑ * → describes the language that contains all strings

∑* 1 → describes the language that contains all the strings ending with 1

Adren's theorem(finite automata to regular expression)

1)



Solution –

Here the initial state and final state is $\mathbf{q_1}$.

The equations for the three states q1, q2, and q3 are as follows –

$q_1 = q_1a + q_3a + ε$ (ε move is because q1 is the initial state0

$q_2 = q_1b + q_2b + q_3b$

$q_3 = q_2a$

Now, we will solve these three equations –

$q_2 = q_1b + q_2b + q_3b$

$= q_1b + q_2b + (q_2a)b$ (Substituting value of q_3)

$= q_1b + q_2(b + ab)$

$= q_1b\ (b + ab)^*$ (Applying Arden's Theorem)

$q_1 = q_1a + q_3a + ε$

$= q_1a + q_2aa + ε$ (Substituting value of q_3)

$= q_1a + q_1b(b + ab^*)aa + ε$ (Substituting value of q_2)

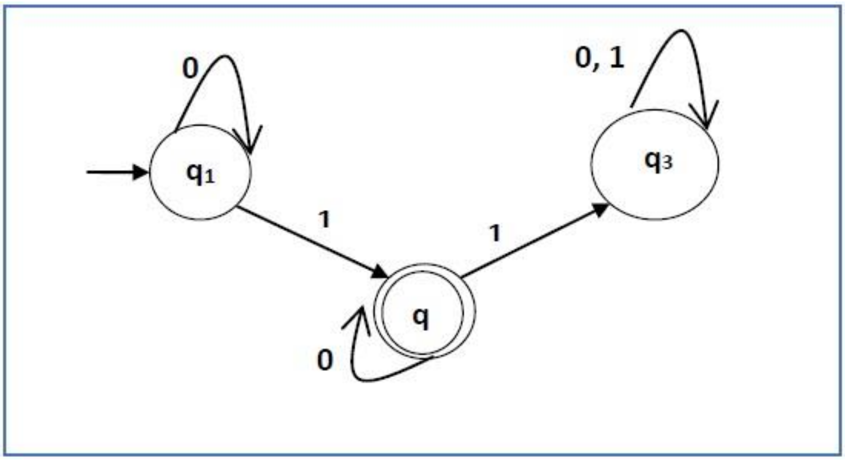
$= q_1(a + b(b + ab)^*aa) + ε$

$= ε\ (a+ b(b + ab)^*aa)^*$

$= (a + b(b + ab)^*aa)^*$

Hence, the regular expression is $(a + b(b + ab)^*aa)^*$.

2)



Solution –

Here the initial state is q_1 and the final state is q_2

Now we write down the equations –

$$q_1 = q_1 0 + \varepsilon$$

$$q_2 = q_1 1 + q_2 0$$

$$q_3 = q_2 1 + q_3 0 + q_3 1$$

Now, we will solve these three equations –

$q_1 = \varepsilon 0^*$ [As, $\varepsilon R = R$]

So, $q_1 = 0^*$

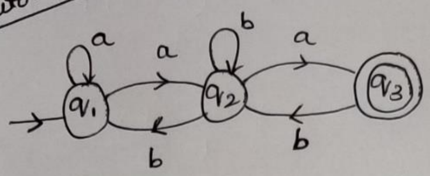
$q_2 = 0^*1 + q_2 0$

So, $q_2 = 0^*1(0)^*$ [By Arden's theorem]

Hence, the regular expression is 0^*10^*.

3) **Question**

Automata



$q_1 \Rightarrow \varepsilon + q_1 a + q_2 b \longrightarrow 1$

$q_2 \Rightarrow q_2 b + q_1 a + q_3 b \longrightarrow 2$

$q_3 \Rightarrow q_2 a \longrightarrow 3$

Substitute 3 in 2

$q_2 \Rightarrow q_2 b + q_1 a + q_2 ab$

$q_2 \Rightarrow q_1 a + q_2 (b+ab)$

$R \Rightarrow Q + R P \Rightarrow QP^*$

$\boxed{q_2 \Rightarrow q_1 a (b+ab)^*} \longrightarrow 4$

Substitute 4 in 1

$q_1 = \varepsilon + q_1 a + (q_1 a (b+ab)^*) b$

$= \varepsilon + q_1 [a + (a(b+ab)^*) b]$

$q_1 = \varepsilon + q_1 [a + (ab(b+ab)^*)]$

$R = Q + R P \Rightarrow QP^*$

$\boxed{q_1 = \varepsilon [a + (ab(b+ab)^*)]^*} \longrightarrow 5$

Substitute 5 in 4

$$q_2 \Rightarrow \left[a + \left(ab\,(b+ab)^*\right)\right]^* a\,(b+ab)^* \rightarrow (6)$$

Substitute (6) in (3)

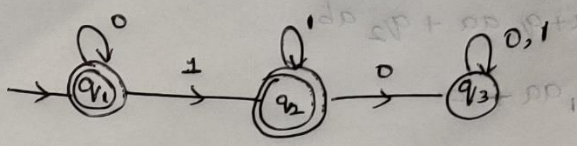
$$q_3 \Rightarrow q_2 a$$

$$q_3 \Rightarrow \left(\left[a + \left(ab\,(b+ab)^*\right)\right]^* a\,(b+ab)^*\right) a$$

The regular expression $\Big\} \left(\left[a + \left(ab\,(b+ab)^*\right)\right]^* a\,(b+ab)^*\right) a$

4) Question

Question



$$\boxed{\begin{aligned} q_1 &\Rightarrow \varepsilon + q_1 0 \\ q_2 &\Rightarrow q_2 1 + q_1 1 \\ q_3 &\Rightarrow q_3 0 + q_3 1 + q_2 0 \end{aligned}}$$

We can write $\begin{pmatrix} q_3 0 \\ + q_3 1 \end{pmatrix}$ as $\boxed{q_3(0+1)}$

$$q_3 \Rightarrow q_3(0+1) + q_2 0 \longrightarrow q_2 0 (0+1)^*$$

$$R \Rightarrow R P + Q \qquad QP^*$$

$$\boxed{q_3 \Rightarrow q_2 0 (0+1)^*}$$

$$q_1 \Rightarrow q_1 0 + \varepsilon \longrightarrow \varepsilon(0)^* \Rightarrow 0^*$$

$$R \Rightarrow RP + Q \longrightarrow QP^*$$

$$\boxed{q_1 \Rightarrow 0^*}$$

$$q_2 \Rightarrow q_2 1 + 0^* 1 \longrightarrow (0^* 1) 1^*$$

$$R \Rightarrow RP + Q \longrightarrow QP^*$$

$$\boxed{q_2 \Rightarrow (0^* 1) 1^*}$$

Regular expression $\Rightarrow (r_1 + r_2)$

$\Rightarrow 0^* + (0^* 1) 1^*$

$\Rightarrow 0^* + 0^* 1 \cdot 1^*$ $\quad (1 \cdot 1^* = 1^*)$

$$\boxed{r \Rightarrow 0^* + 0^* 1^*}$$