AFLL

UNIT - 2

CLASS NOTES

feedback/corrections: vibha@pesu.pes.edu

Vibha Masti

Regular
EXPRESSIONS

Representing Regular Languages

- DFA
- NFA
- λ-NFA
- Regular expressions

Regular Expression

- pattern that contains a sequence of characters that is matched against some text

1) Atoms
   - what you are looking for
   - where in the text
2) Operators
   - combine atom & operators – complex expression

Atoms

$(A) \Rightarrow \Sigma \longrightarrow \mathcal{L}(A) = \{A\}$

$\phi \rightarrow$ RE that denotes empty language (set)

$\mathcal{L}(\phi) = \{\}$

$\lambda / \epsilon \rightarrow$ RE that denotes empty string

$\mathcal{L}(\lambda) = \{\lambda\}$

Operators

R_1 and R_2 — L_1 and L_2

1) Union (+)

$R_1 + R_2$ — $L_1 \cup L_2$ (order does not matter)

2) Concatenation (·)

$R_1 \cdot R_2$ — $L_1 \cdot L_2$ (order matters)
L_1 followed by L_2

3) Closure (*)

R^* — L^* (0 or more times)

4) Incomplete closure (+)

R^+ — L^+ (1 or more times)

Precedence

1) Brackets ( )
2) Closure *, +
3) Concat ·
4) Union +

Algebraic Laws

Identity for operator

operator applied to identity & another value, result is the other value

Question 1

Construct RE over $\Sigma = \{a, b\}$ that accepts a string that starts with ab

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

Question 2

Construct RE over $\Sigma = \{a, b\}$ that accepts a string that contains a substring ab

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

Question 3

Construct RE over $\Sigma = \{a, b\}$ that accepts a string that ends with ab

concatenation; dot not req.

$$L = (a+b)^* \cdot ab$$

Question 4

RE that ends with ab or ba

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

Question 5

RE, starts & ends with same symbol $\Sigma = \{a,b\}$

Case 1: $a \_\_ a$

or

Case 2: $b \_\_ b$

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

Question 6

RE, $\Sigma = \{a,b\}$, starts & ends with diff symbols

$$L = a(a+b)^*b \; + \; b(a+b)^*a$$

Question 7

Third symbol from the start is a (left)

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

Question 8

2^{nd} symbol from right (end) is a

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

Question 9

3^{rd} symbol from right (end) is a and 4^{th} symbol from right (end) is b

$$L = (a+b)^* ba (a+b)(a+b)$$

Question 10

At least one a and at least one b

$$L = \{ab, ba \ldots\}$$

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

Question 11

At least one a in the first 3 symbols and at least one b in the last two symbols

min length: 3, but we take 5

$$\Big([a(a+b)(a+b)] + [(a+b)a(a+b)] + [(a+b)(a+b)a]\Big) \cdot (a+b)^* \cdot$$

$$\cdot [(a+b)b + b(a+b)]$$

Question 12

Length is exactly 2, $\Sigma = \{a, b\}$

$$L = (a+b)(a+b) \quad \text{OR} \quad (a+b)^2$$

Question 13

$|w| \geq 2$, $w \in \{a,b\}^*$

$$L = (a+b)^2 (a+b)^*$$

Question 14

$|w| \leq 2$, $w \in \Sigma^*$, $\Sigma = \{a, b\}$

$$L = \{\lambda, (a+b), (a+b)^2\}$$

$$L = \lambda + (a+b) + (a+b)^2$$

OR

$$L = (\lambda + a + b)(\lambda + a + b)$$

$$L = (\lambda + a + b)^2$$

Generic — Lengths

$|w| = n, \quad L = (a+b)^n$

$|w| \geq n, \quad L = (a+b)^n (a+b)^*$

$|w| \leq n, \quad L = (\lambda + a + b)^n$

Question 15

$|w| \bmod 2 = 0$

$$L = [(a+b)^2]^*$$

Question 16

$|w| \bmod 2 = 1$

$$L = (a+b)[(a+b)^2]^*$$

Question 17

$|w| \bmod 3 = 0$

$$L = ((a+b)^3)^*$$

Question 18

$|w| \bmod 3 = 2$

$$L = (a+b)^2((a+b)^3)^*$$

Question 19

$$L = \{ n_a(w) = 2,\ w \in \{a,b\}^* \}$$

$$L = b^* a b^* a b^*$$

Question 20

$$L = \{ n_a(w) \geq 2,\ w \in \{a,b\}^* \}$$

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

Question 21

$$L = \{ n_a(w) \leq 2,\ w \in \{a,b\}^* \}$$

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

Question 22

$$n_a(w) \bmod 2 = 0$$

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

Question 23

$$n_a(w) \bmod 2 = 1$$

$$(b^* a b^* a b^*)^* b^* a b^*$$

Question 24

* $n_a(w) \bmod 3 = 0$

$b^*(b^*ab^*ab^*ab^*)^*b^*$ or $(b^*ab^*ab^*ab^*)^* + b^*$

Question 25

$n_a(w) \bmod 3 = 2$

$(b^*ab^*ab^*ab^*)^*\ b^*ab^*ab^*$

Question 26

note: a^nb^n: impossible as DFA/RE

$L = \{a^n b^m \mid n+m \text{ is even}\}$

even a's   even b's
odd a's   odd b's

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

Question 27

$L = \{a^{2n}b^{2m} \mid n,m \geq 1\}$

$(aa)^+(bb)^+$

Question 28

$$L = \{a^n b^m \mid n, m \geq 1,\ \text{product}(n,m) \geq 3\}$$

	No. of a's	No. of b's	≥ 3
Case 1	1	≥ 3	≥ 3
Case 2	≥ 3	1	≥ 3
Case 3	≥ 2	≥ 2	≥ 3

$$aa^*bbbb^* + aaa^*bbb^* + aaaa^*bb^*$$

Question 29

$$L = \{a^n b^m \mid n \geq 4,\ m \leq 3\}$$

$$aaaaa^*(\lambda + b + bb + bbb)$$

or

$$aaaaa^*(\lambda + b)^3$$

Question 30

$$L = \{a^n b^m c^k \mid n+m \text{ is odd and } k \text{ is even}\}$$

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

Question 31

$$L = \{a^n b^m c^k \mid n \leq 4,\ m \geq 2,\ k \leq 2\}$$

$$(\lambda + a)^4 (bb) b^* (\lambda + c)^2$$

Question 32

6 6

$$L = \{ aba,\ a^5,\ aba^7,\ a^{11},\ aba^{13} \}$$

6

$$(aba + a^5)(a^6)^*$$

Question 33

$$L = \{ uvw \mid u, v, w \in \{a, b\}^*,\ |u| = 2,\ |w| = 2 \}$$

$$(a+b)^2 (a+b)^* (a+b)^2$$

Question 34

$L = \{ w \mid |w|$ divisible by 2, not by 3 $\}$

$$\left((a+b)^2 + (a+b)^4\right)\left((a+b)^6\right)^*$$

Question 35

Even no. of a's followed by even/odd b's

$$L = (aa)^* b^*$$

Question 36

Begin with at least 2 a's, end with even b's

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

Question 37

* Accepts binary string such that if dec. eq. of the no. is even then the length of the string is even, if dec. eq is odd then length of string is odd

$$((0+1)^2)^*(0+1)0 + ((0+1)^2)^* 1$$

Question 38

$L = \{$Binary string whose decimal eq. is twice an odd no$\}$

*

2×odd	Dec.	Bin
2×1	2	$(10)_2$
2×3	6	(110)
2×5	10	$(1010)_2$
2×7	14	$(1110)_2$

ends with 10

$$(0+1)^* 10$$

Question 39

Bin string that does not have 2 consecutive 0s

$$(11^*0)^* + (011^*)^* + 1^* + 01^*$$

Question 40

Exactly one pair of consecutive 0s

$$(01+1)^* 00 (10+1)^*$$

Question 41

No consecutive a's and bs

$$(b+\lambda)(ab)^*(a+\lambda) + (a+\lambda)(ba)^*(b+\lambda)$$

Question 42

What is L for $(a+b)^* aa (a+b)^*$

strings with at least one pair (aa)

Equivalence of Regular Expressions

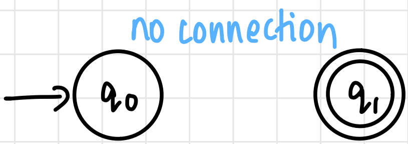
FA RE (λ-NFA)

- Thompson construction method : RE $\rightarrow$ FA
- State elimination method : FA $\rightarrow$ RE
- Kleene's Theorem

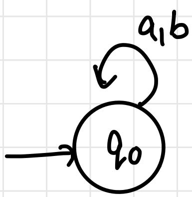
RE to FA

- RE to λ-NFA

1) RE accepts empty language ϕ

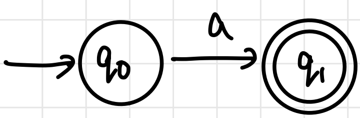


for $\Sigma = \{a, b\}$



2) RE accepts λ (empty string)

3) RE is single symbol a



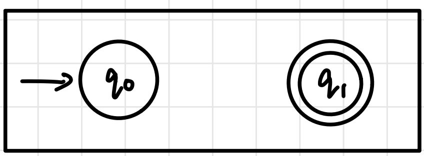
Without loss of generality,

R_1 and R_2

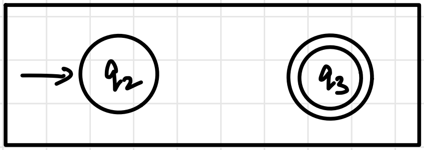
union	$R_1 + R_2$	$\longrightarrow$	$L_1 \cup L_2$
concat	$R_1 R_2$	$\longrightarrow$	$L_1 L_2$
closure	$R_1{}^*$	$\longrightarrow$	$L_1{}^*$

UNION

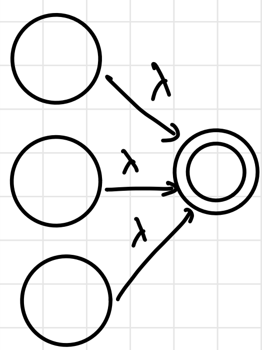
R_1



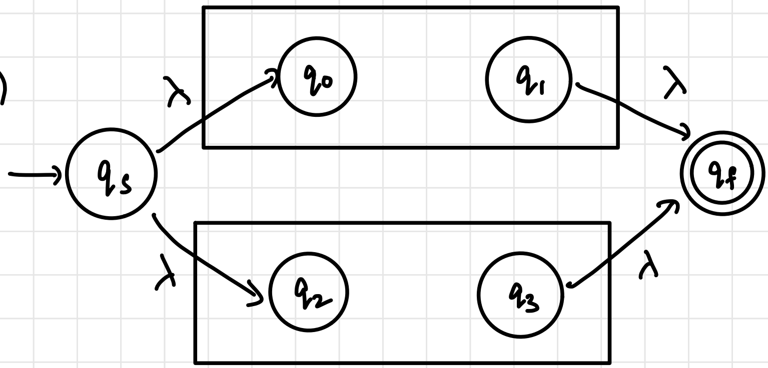
R_2



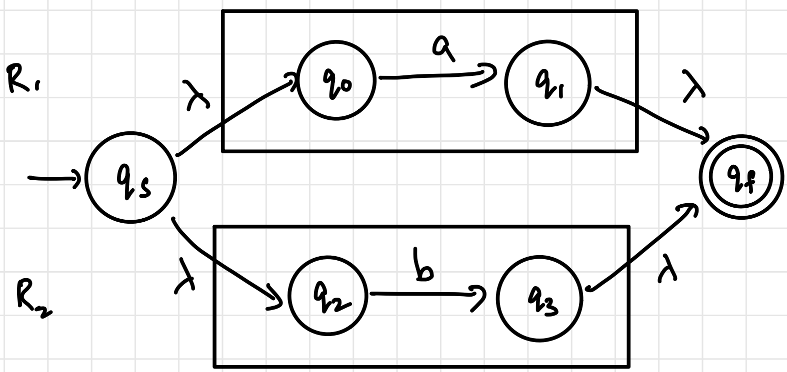
for > 1 FS



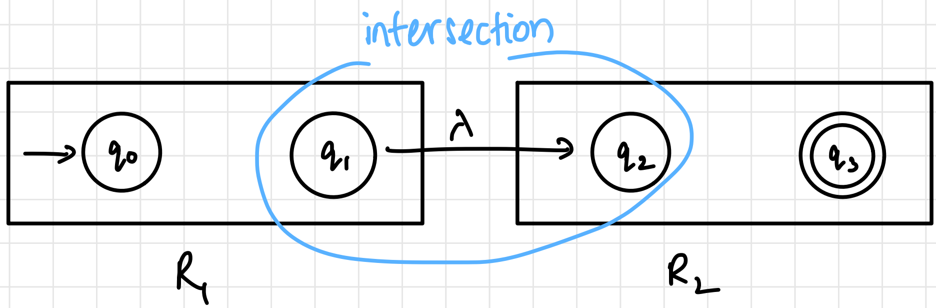
$R_1 + R_2$
(union)



eg: $(a+b)$

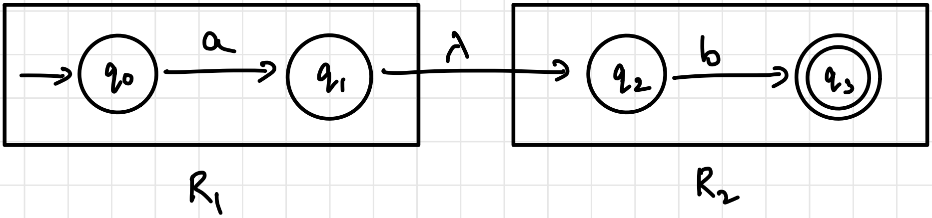


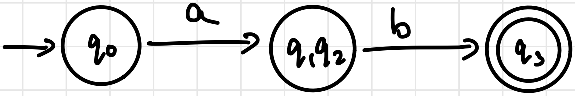
CONCATENATION



$R_1 R_2$ (concatenation)

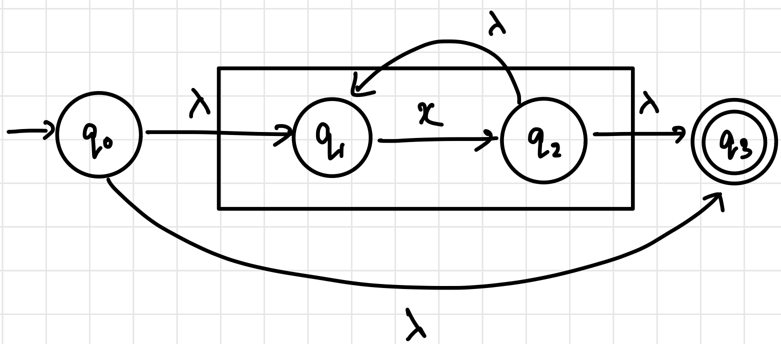
eg: (ab)





CLOSURE

$R_1^* \longrightarrow L_1^* = L_1^0 \cup L_1^1 \cup L_1^2 \cup \dots$



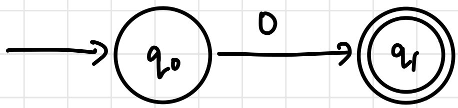
R_1^*

for L_1^*

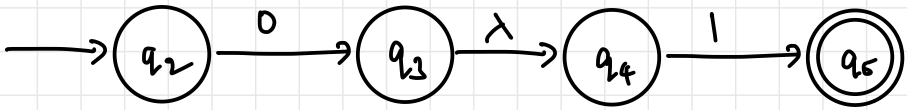
Question 43

convert $(0+01)^*$ using Thompson method

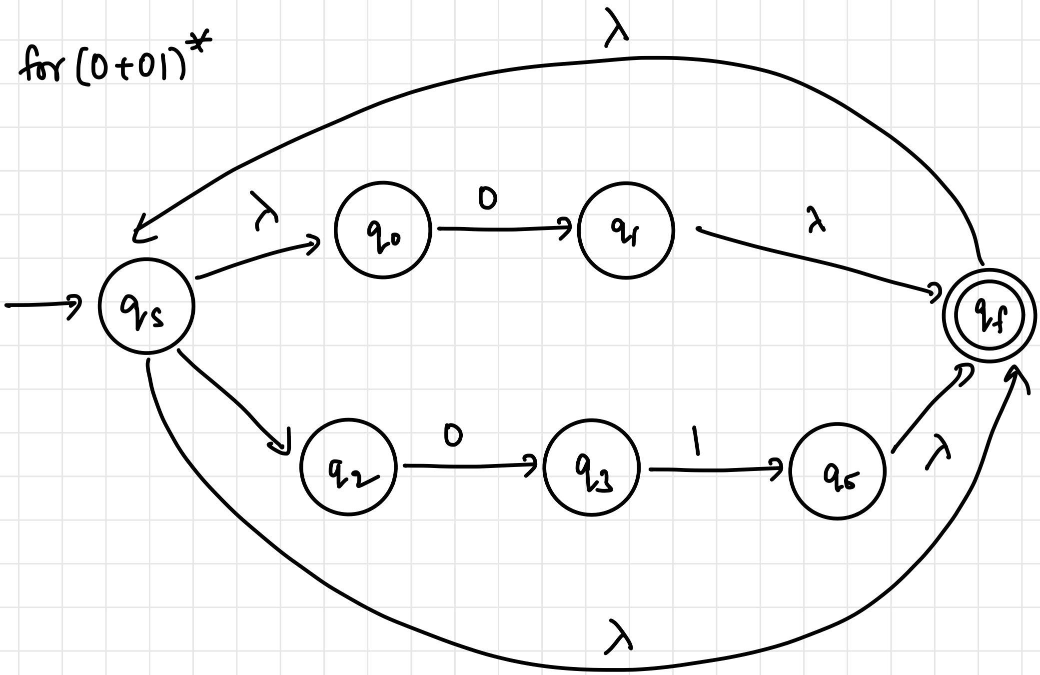
for 0



for 01

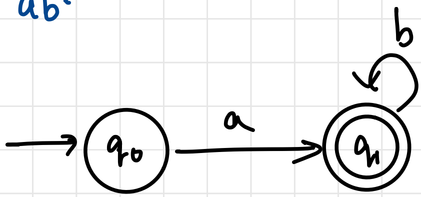


for $(0+01)^*$



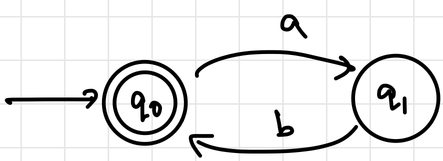
Question 44

ab^*



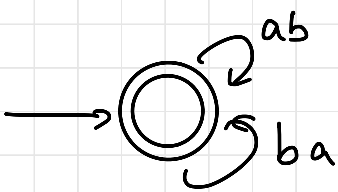
Question 45

$(ab)^*$

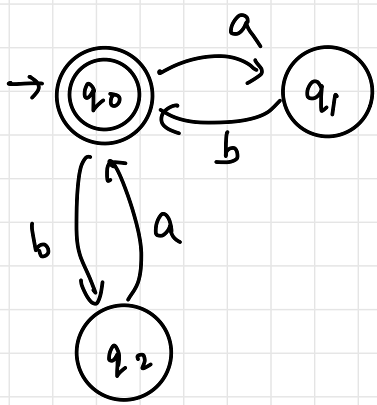


Question 46

$(ab + ba)^*$

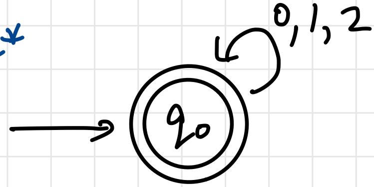


expand



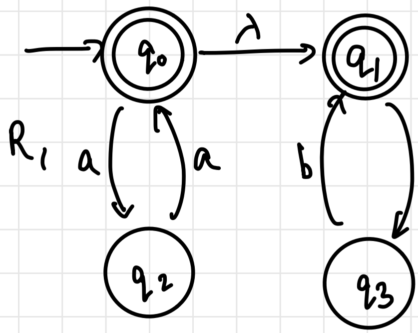
Question 47

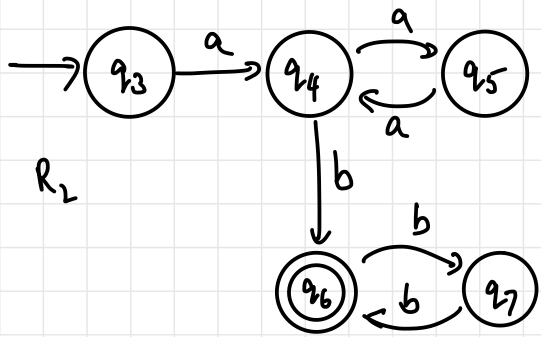
$0^* + 1^* + 2^*$



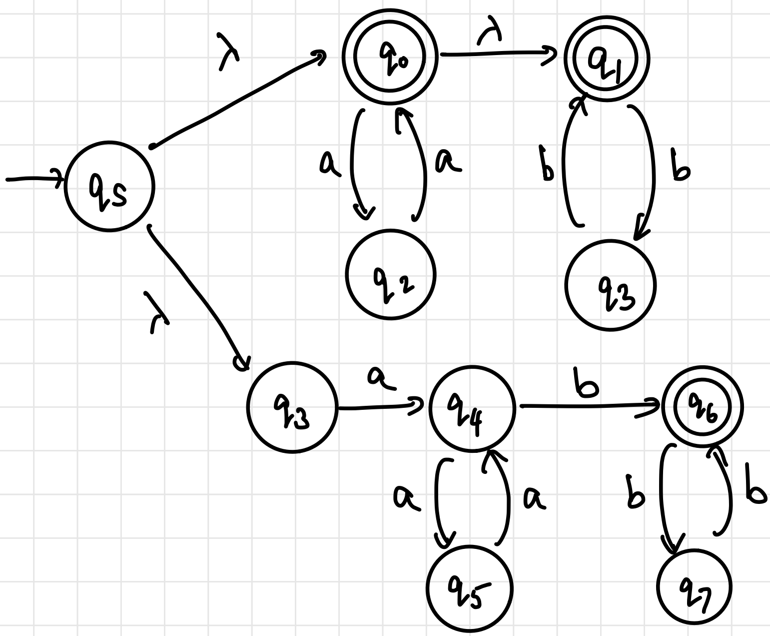
Question 48

$R_1 \; (aa)^*(bb)^* + a(aa)^*b(bb)^* \; R_2$



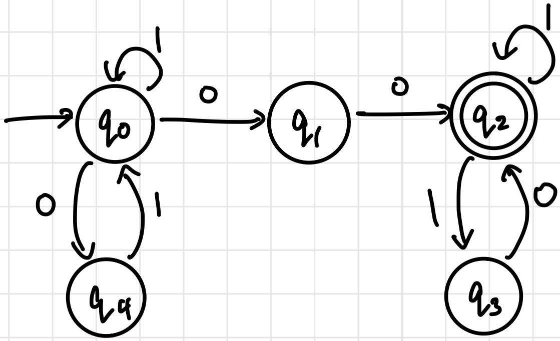


$R_1 + R_2$



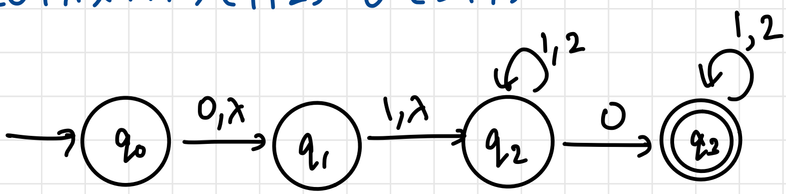
Question 49

$$(1+01)^* \, 00 \, (1+10)^*$$



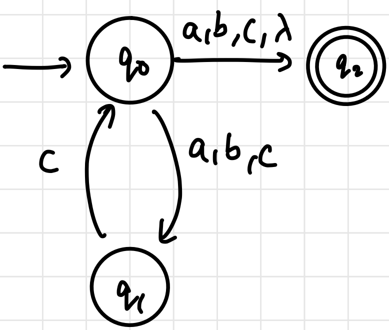
Question 50

$$(0+\lambda)(1+\lambda)(1+2)^* \, 0 \, (2+1)^*$$



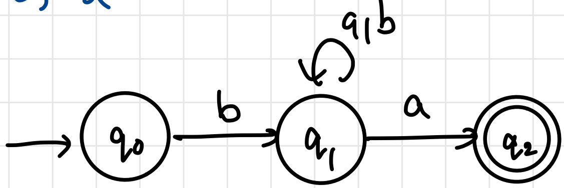
Question 51

$$((a+b+c)c)^* (a+b+c+\lambda)$$



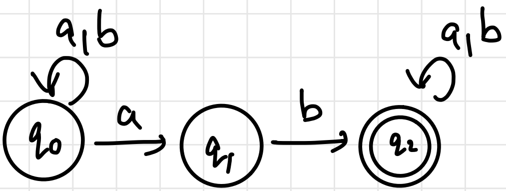
Question 52

$b(a+b)^*a$



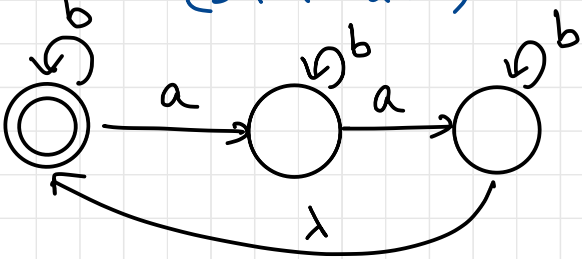
Question 53

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



Question 54

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

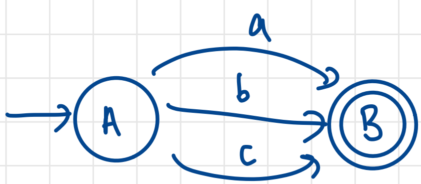


FA to RE

State reduction / state elimination method

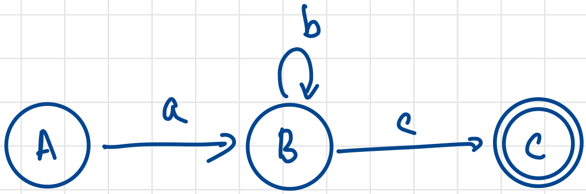
- Do not eliminate start & final state
- Start state having incoming edge, introduce new start state, make transition from NSS
- Final state having outgoing edge, introduce new final state
- Multiple final states, introduce new single FS.

Question 55



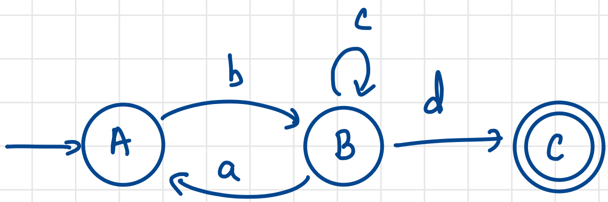
$(a+b+c)$

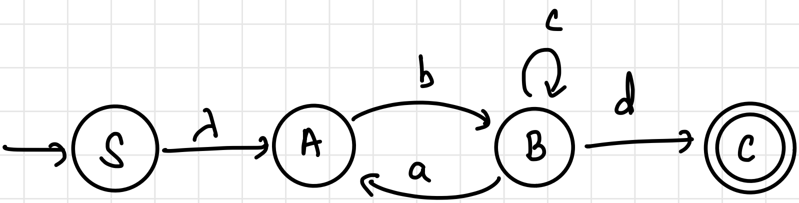
Question 56



ab^*c

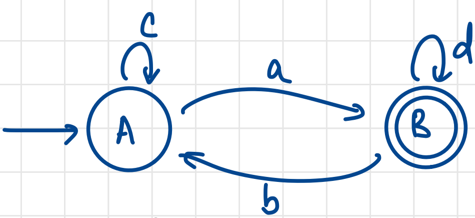
Question 57

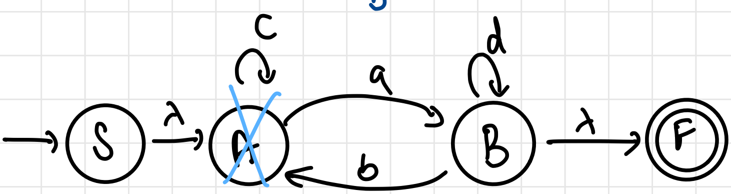


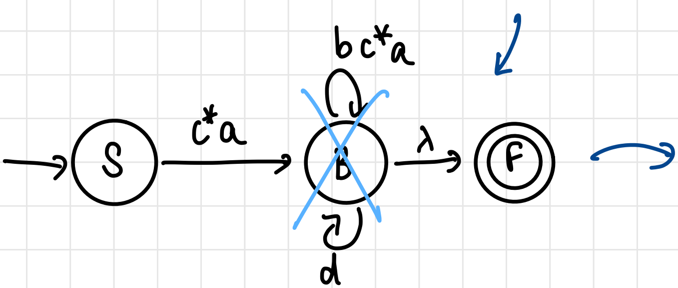


$bc^*(ab+c)^*d$ or $(bc^*a)^*bc^*d$

Question 58

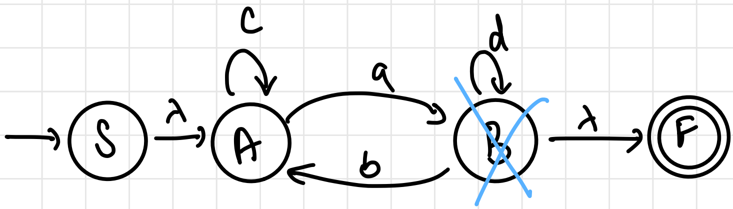


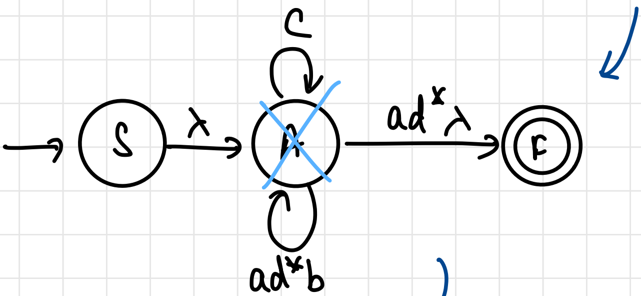




$c^*a(bc^*a+d)^*$

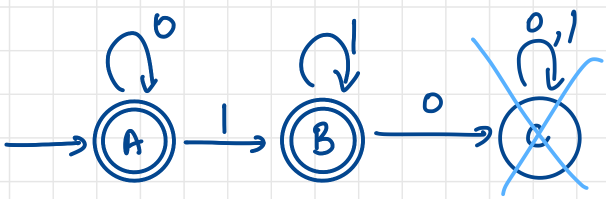
OR

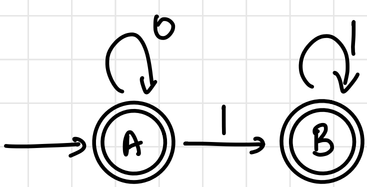


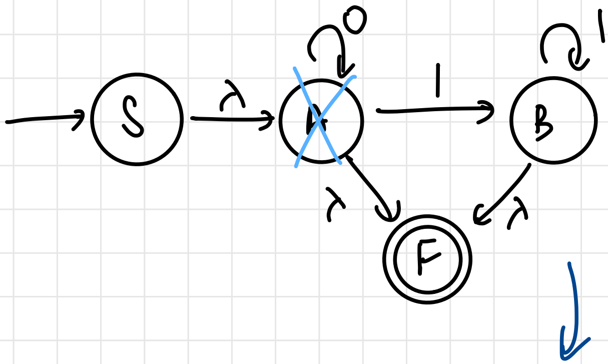


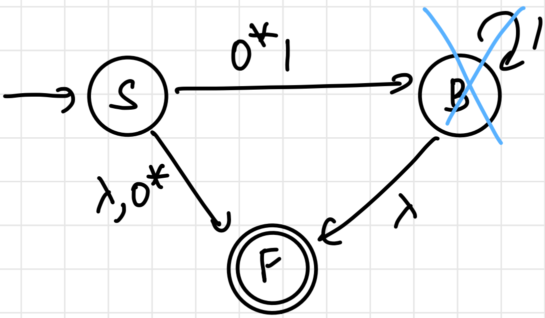
$(ad^*b+c)^* ad^*$

Question 59



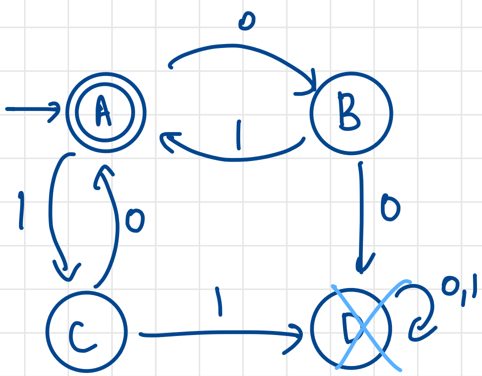


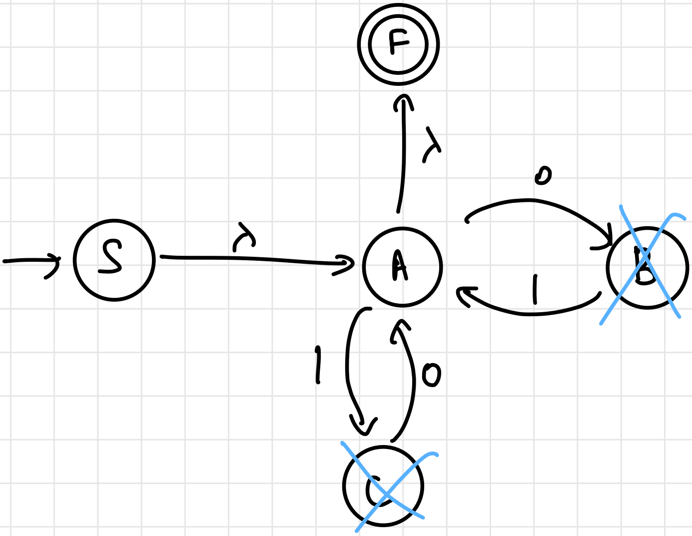


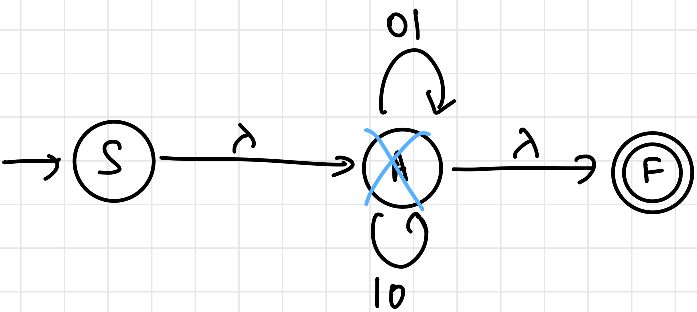


$$(0^*11^* + 0^* + \lambda) = 0^* + 0^*11^*$$

Question 60

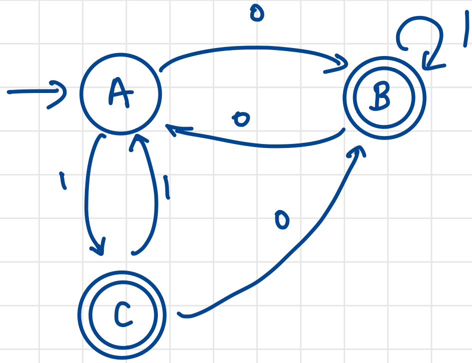






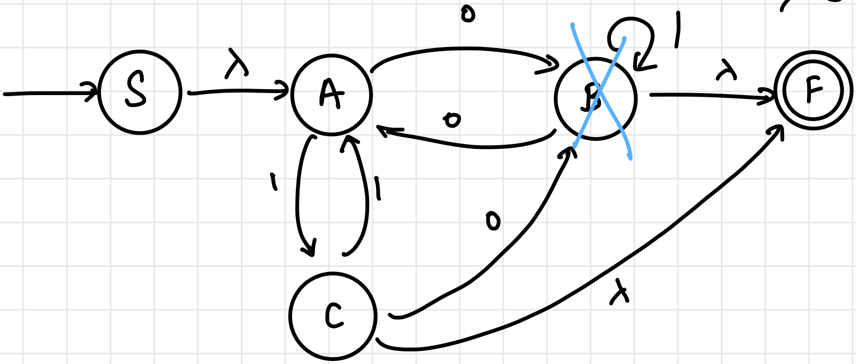
$(01+10)^*$

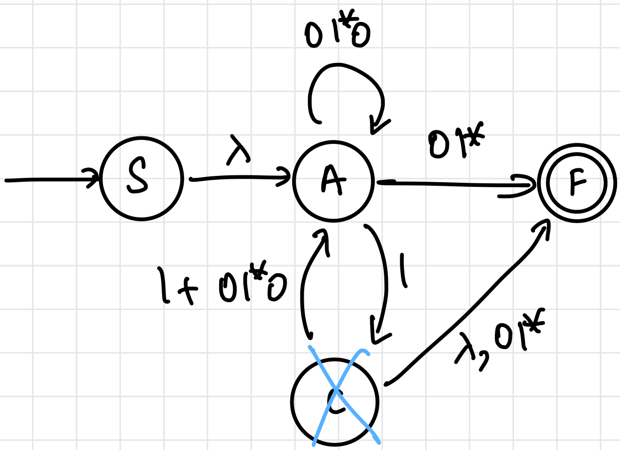
Question 61

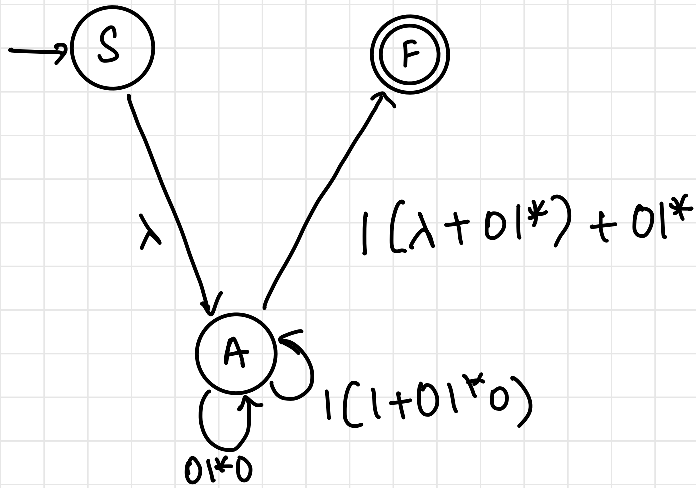


$\lambda 0 1^* \lambda$

$\lambda \cup 1^*(00)^* 1^* \lambda$

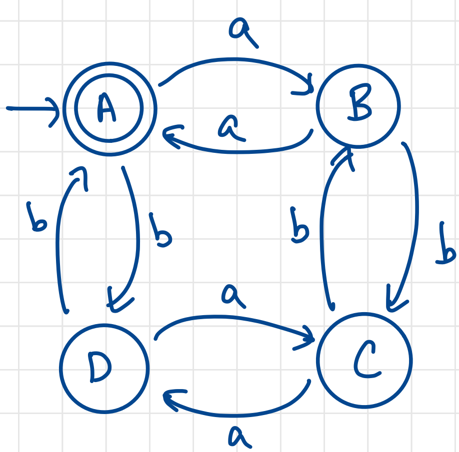


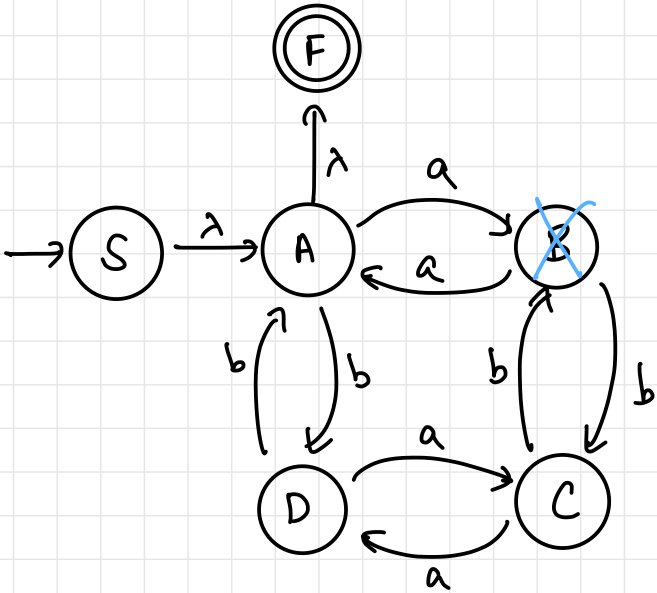


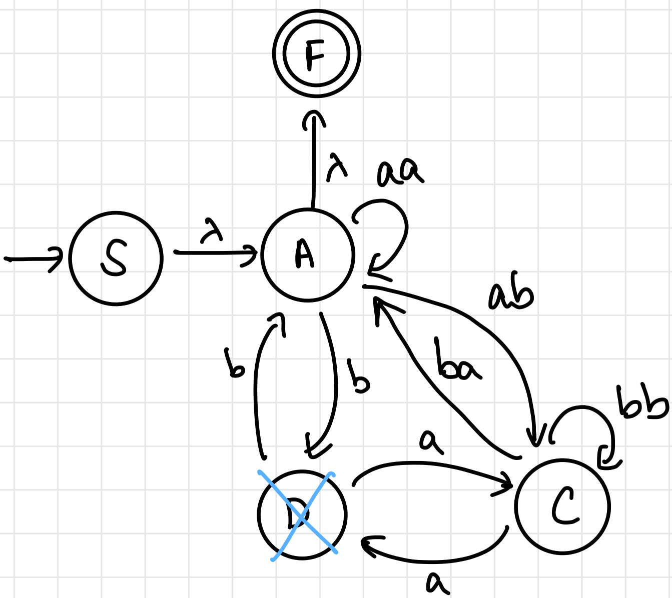


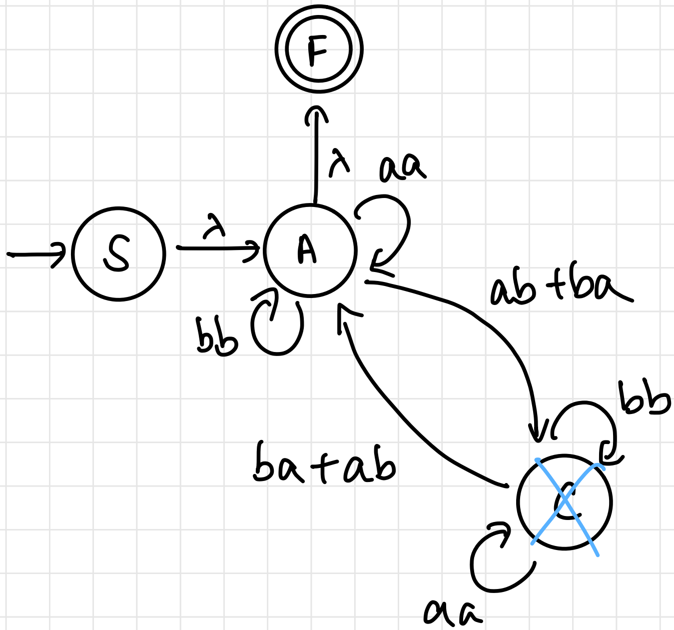
$$(01^*0+1(1+01^*0))^*\ (1(\lambda+01^*)+01^*)$$

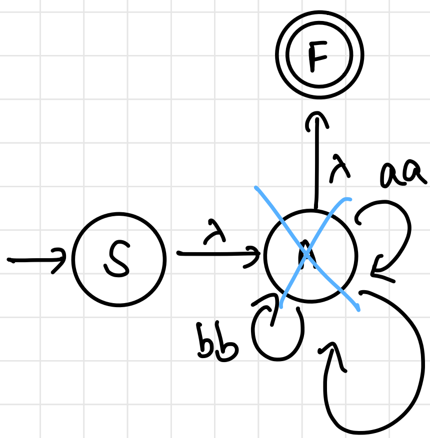
Question 62




F
S
A
B
C
D
λ
a
b


F
S
A
C
D
λ
aa
ab
ba
bb
a
b





$(ab+ba)(bb+aa)^*(ba+ab)$

$$\left(aa+bb+\left[(ab+ba)(bb+aa)^*(ab+ba)\right]\right)^*$$

Equivalence of RE

Examples

Question 63

$$0^* + 0^*11^* \equiv 0^*(\lambda + 11^*)$$

$$0^*(\lambda + 11^*) \equiv 0^*(\lambda + 11^*)$$

$$0^*(\lambda + 1^+) \equiv 0^*(\lambda + 1^+)$$

$$0^*1^* \equiv 0^*1^*$$

Question 64

$(1+00^*1) + (1+00^*1)(0+10^*1)^*(0+10^*1)$ is equal to $0^*1(0+10^*1)^*$

$$(1+00^*1) + (1+00^*1)(0+10^*1)^*(0+10^*1) \equiv 0^*1(0+10^*1)^*$$

$$(1+00^*1)\left(\lambda + (0+10^*1)^*(0+10^*1)\right) \equiv 0^*1(0+10^*1)^*$$

$$(1+00^*1)(\lambda + (0+10^*1)^+) \equiv 0^*1(0+10^*1)^*$$

$$(1+00^*1)(0+10^*1)^* \equiv 0^*1(0+10^*1)^*$$

$$(\lambda + 00^*)1(0+10^*1)^* \equiv 0^*1(0+10^*1)^*$$

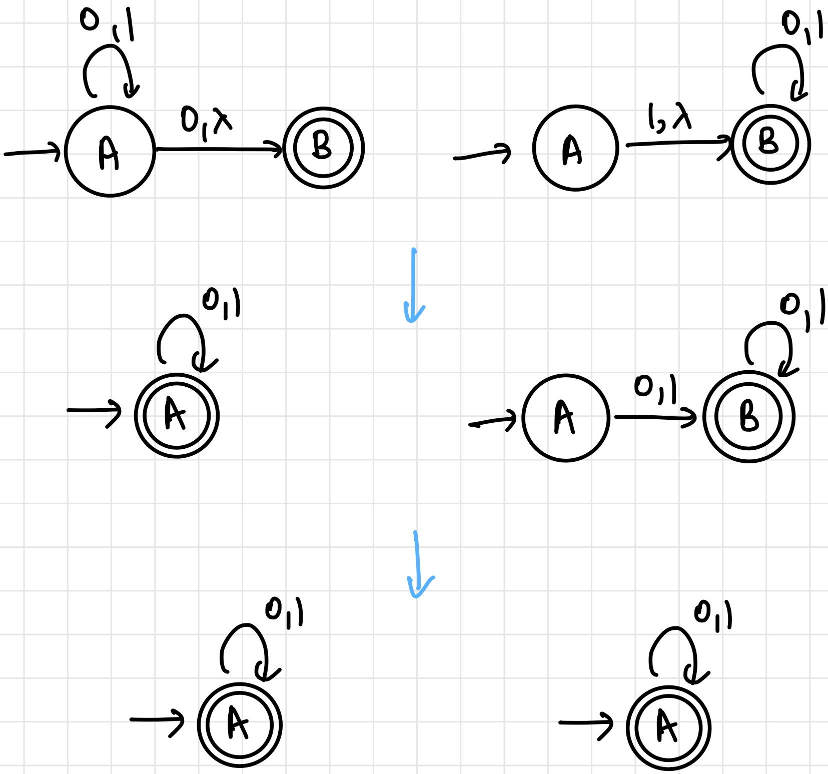
$$0^*1(0+10^*1)^* \equiv 0^*1(0+10^*1)^*$$

Formal Method

- Convert REs to NFAs, then to DFAs, then compare equality using table filling algorithm

Question 65

$(0+1)^*(0+\lambda)$ and $(1+\lambda)(1+0)^*$



Regular Expressions in Practice

- Practical / real life strings
- pan card no, aadhaar card, date -2020, email
- packages in Python (re), Java etc.
- practical symbols / special characters meta characters

Characters

1) dot (.) – period operator
   - matches a single character, no matter what that char is, except newline
   - eg: a. ← any char follows a

2) star (*)
   - preceding character 0 or more times
   - eg: a^* $\{\lambda, a, aa, aaa \dots\}$

3) Plus (+)
   - preceding char one or more times
   - eg: a^+ $\{a, aa, aaa \dots\}$

4) Question mark (?)
   - preceding character upto one time
   - eg: xy?z {xyz, xz}

5) Square brackets ([ ]) - character class

- range specified in [ ]
- matches a single character in [ ]

- eg: [x y z] x or y or z

  [0-9] 0 or 1 or . . . or 9

  [a-z A-Z 0-9]

6) Cap/Carrot Symbol (^)

(i) anchor for the start (starts with)

- eg: ^abc , string starts with abc

(ii) negation

- eg: [^abc] except a or b or c, any symbol (could be xyef...)

7) Dollar ($)

- anchor for the end (ends with)

- eg: efg$, ends with efg

Note: ^$: empty string

8) Curly braces ({m,n})

- repetition from m to n times

- eg: a{2,4} a 2, 3 or 4 times

- eg: $a\{0,\} \rightarrow a^*$

  $a\{1,\} \rightarrow a^+$

  $a\{2,3\} \rightarrow$ aa or aaa

  $a\{2\} \rightarrow$ aa

Pan Card Number

- 10 characters
- first 5 chars, uppercase from A to Z
- next 4 chars, any digit from 0 to 9
- last character, any uppercase letter

Regular Expression

can be \d

1) ^[A-Z]{5}[0-9]{4}[A-Z]$

2) ^[A-Z]{5}\d{4}[A-Z]$

Indian Mobile Number

- 10 digit number
- starts with 9, 8, 7, 6
- next 9 : from 0 to 9
- optional 0 or +91

or

[0|+91]?[6-9][0-9]{9}

Aadhaar Number

- 12 digit number
- Space after every four digits
- first digit is not 0 or 1

[2-9][0-9]{3}\s [0-9]{4}\s[0-9]{4}

Date in the year 2020

1 3 5 7 8 10 12

31 days — Jan, Mar, May, July, Aug, Oct, Dec

4 6 9 11

30 days — Apr, Jun, Sep, Nov

29 days — Feb 2

dd-mm-yyyy or dd/mm/yyyy or dd.mm.yyyy

```
[((0[1-9]|[10-31]) [/-.] (0[13578]|1[02]))|
 ((0[1-9]|[10-30]) [/-.] (0[469]|11))|
 ((0[1-9]|[10-29]) [/-.] (02))] - 2020$
```

Email Address

personal_info @ domain_name (only .com & .org)

$[a\text{-}z\ A\text{-}Z\ 1\text{-}9\ .\ \_]^+ @ [a\text{-}z\ A\text{-}Z\ 1\text{-}9\ \_].[com|org]$

Grammar

$G = \{V, T, P, S\}$

V = set of variables (non-terminals) → capital (states)

T = set of terminals (symbols in the alphabet, end string)

P = production rules (rules of grammar)

P ($\alpha \to B$)

α: non-terminal

B: string over $(V \cup T)^*$

S = start symbol → (S)

1) Linear grammar

$\underset{LHS}{\alpha} \to \underset{RHS}{\beta}$

at most one non-terminal at the RHS of the production

$S \to aSb \mid bSa \mid Sa \mid aS \mid \lambda \mid aa$

2) Non-linear grammar

no restrictions

$$S \to aSS \,/\, aSSb \,/\, SSa \,/\, Sa \,/\, AB$$

Regular Grammar $\subset$ Linear Grammar

(i) Left linear Grammar

if nonterminal is leftmost symbol

$$S \to Sa \mid \lambda$$

(ii) Right Linear Grammar

if nonterminal is rightmost symbol

$$S \to aS \mid \lambda$$

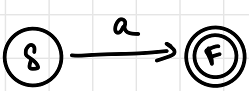
Automata to Grammar - Right Linear

Question 66

$L = \{a\}$, write RG for RE = a

RG:

$$S \to a$$



Question 67

$L = \{aaaa\}$, write RG RE = a^4

RG: $S \to aaaa$

Question 68

$L = \{\lambda, a, aa, aaa \ldots\}$

$RE = a^*$

RG:

$S \to aS \mid \lambda$

eg: aaaa

Derivation

$$
\begin{aligned}
S &\Rightarrow aS \\
&\Rightarrow a\,aS \\
&\Rightarrow aa\,aS \\
&\Rightarrow aaaaS \\
&\Rightarrow aaaa\lambda
\end{aligned}
$$

Question 69

$L = \{a, aa \ldots\}$ $RE = a^+$

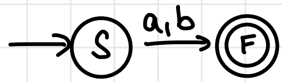
RG: $S \to aS \mid a = a^+$

eg: aaaa

$$
\begin{aligned}
S &\Rightarrow aS \\
&\Rightarrow aaS \\
&\Rightarrow aaaS \\
&\Rightarrow aaaa
\end{aligned}
$$

Question 70

$(a+b)$

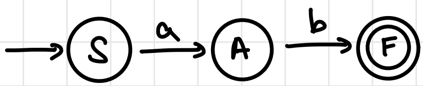


$S \to a \mid b$

Question 71

ab

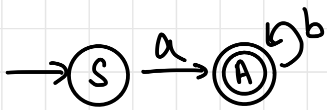
$S \to ab$



$S \to aA$
$A \to b$

Question 72

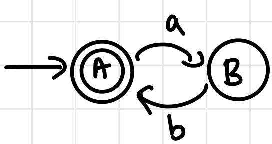
ab^*



$S \to aA \mid \lambda$
$A \to bA \mid b$

Question 73

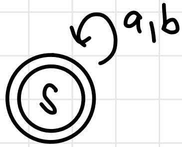
$(ab)^*$



$S \to aA \mid \lambda$
$A \to bS$

Question 74

$(a+b)^*$



$S \to aS \mid bS \mid \lambda$

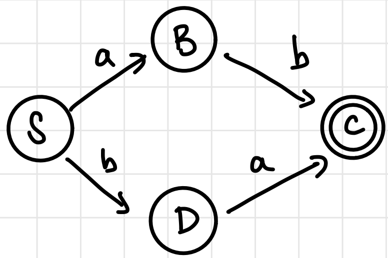
Question 75

$(a+b)^+$

$S \to aS \mid bS \mid a \mid b$

Question 76

$ab+ba$



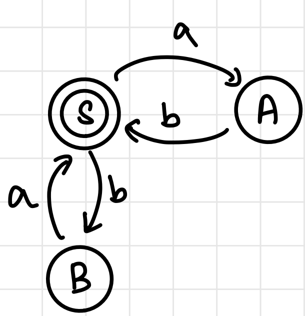
$S \to ab \mid ba$

OR

$S \to aB \mid bD$
$B \to b$
$D \to a$

Question 77

$(ab+ba)^*$



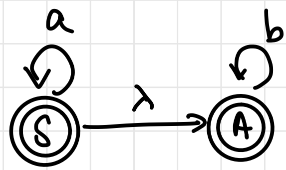
$S \to abS \mid baS \mid \lambda$

OR

$S \to aA \mid bB \mid \lambda$
$A \to bS$
$B \to aS$

Question 78

$a^* b^*$



$S \to aS \mid A$
$A \to bA \mid \lambda$

eg: aabb

$$\begin{aligned} S &\Rightarrow aS \\ &\Rightarrow aaS \\ &\Rightarrow aaA \\ &\Rightarrow aabA \\ &\Rightarrow aabbA \\ &\Rightarrow aabb\lambda \end{aligned}$$

Question 79

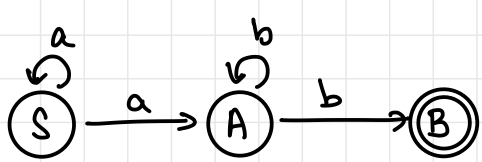
$(ab+ba)^+$

$S \to abS \mid baS \mid ab \mid ba$

Question 80

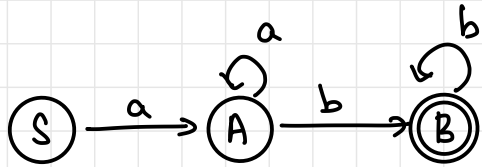
$L = \{a^n b^m \ (n, m \geq 1)\}$; RE = $a^+ b^+$

$S \to aS \mid aA$
$A \to bA \mid b$



OR

$S \to aA$
$A \to aA \mid bB$
$B \to bB \mid \lambda$

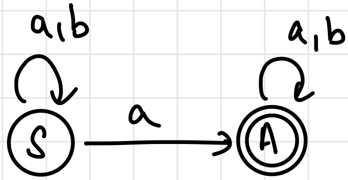


Question 81

$L = \{\text{at least one } a\}$

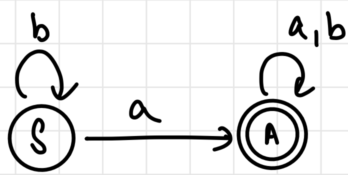
$(a+b)^* a (a+b)^*$

$S \to aS \mid bS \mid aA$
$A \to aA \mid bA \mid \lambda$



OR

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

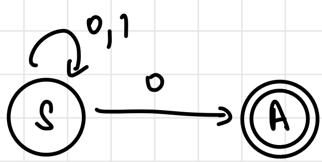


$S \to bS \mid aA$
$A \to aA \mid bA \mid \lambda$

Question 82

Binary even no. $= (0+1)^* 0$

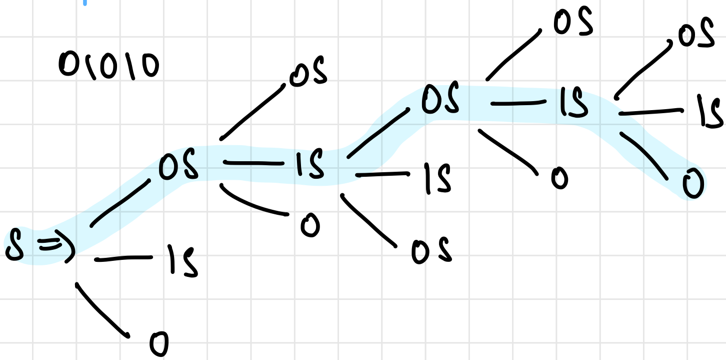
$S \to 0S \mid 1S \mid 0$



Derivation using tree

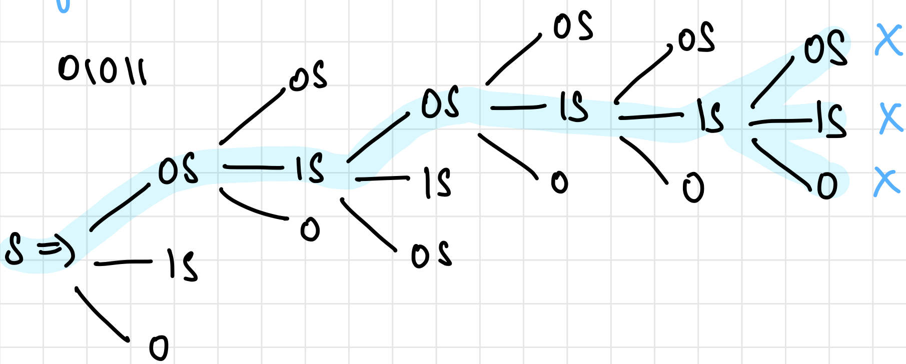
(i) Accepted

01010

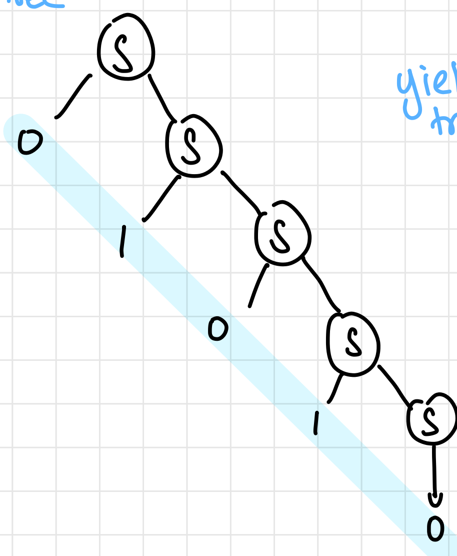


(ii) Rejected

01011



Parse tree

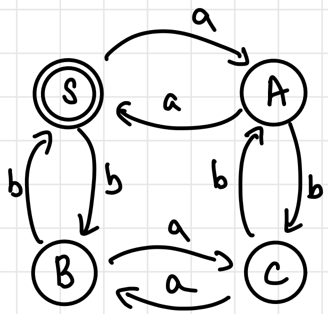


yield of tree = 01010

Question 83

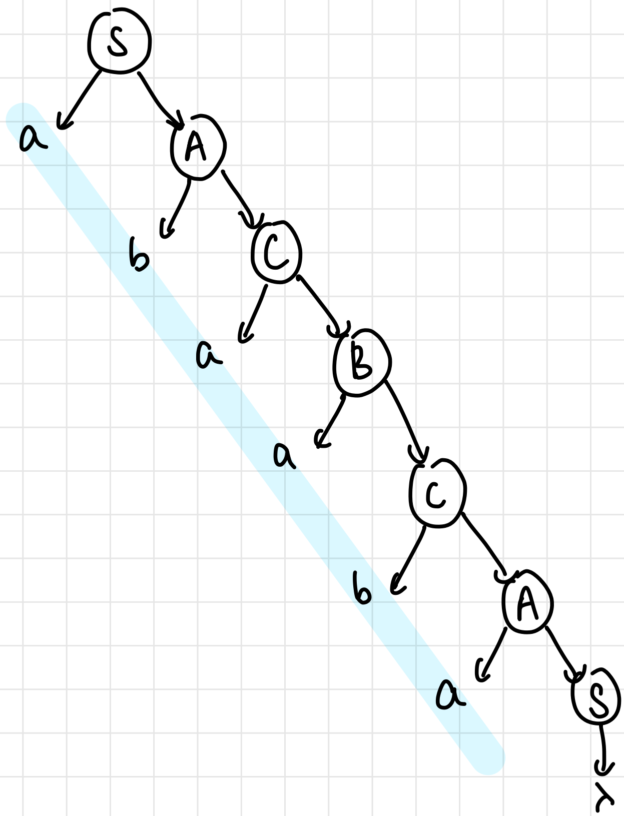
$$L = \{ n_a(w) \bmod 2 = 0 \text{ and } n_b(w) \bmod 2 = 0 \}$$

$S \to aA \mid bB \mid \lambda$
$A \to aS \mid bC$
$B \to bS \mid aC$
$C \to aB \mid bA$



Parse tree

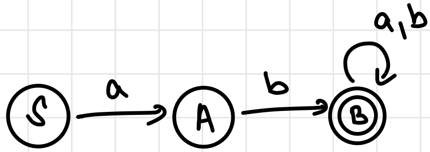
abaaba



Question 84

$L = \{abw,\ w \in \{a,b\}^*\}$

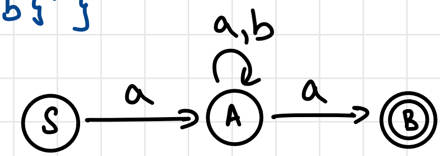
$S \to aA$
$A \to bB$
$B \to aB \mid bB \mid \lambda$

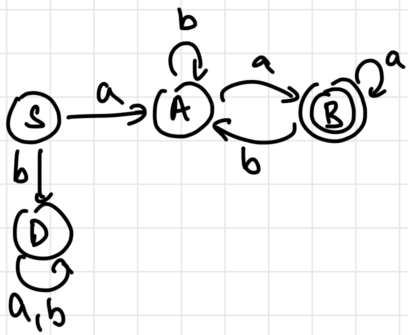


Question 85

$L = \{awa,\ w \in \{a,b\}^*\}$

$S \to aA$
$A \to aA \mid bA \mid a$

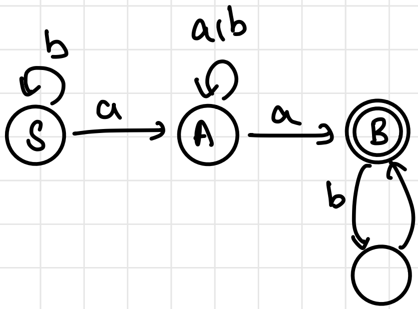




Grammar to Automata

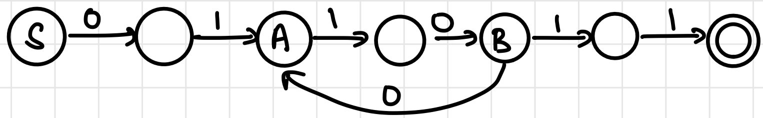
Question 86

$S \to bS \mid aA$
$A \to aA \mid bA \mid aB$
$B \to bbB$
$B \to \lambda$



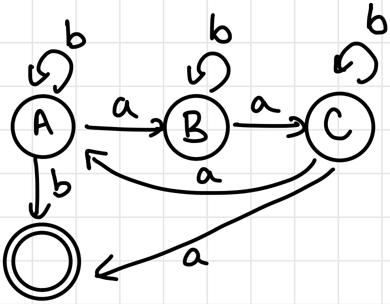
Question 87

$S \rightarrow 01A$
$A \rightarrow 10B$
$B \rightarrow 0A \mid 11$



Question 88

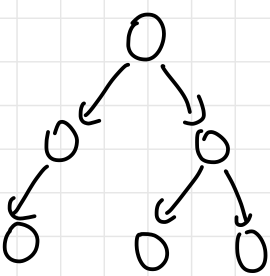
$A \rightarrow aB \mid bA \mid b$
$B \rightarrow aC \mid bB$
$C \rightarrow aA \mid bC \mid a$



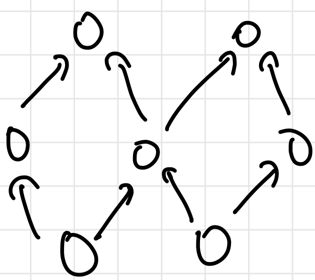
Parsing

Parsing is a process of determining whether a string $w \in L(G)$

1) Top-down parsing



2) Bottom-up parsing



Parsing from Right Linear Grammar → Rightmost derivation

$S \rightarrow 0S \mid 1S \mid 0$

1010

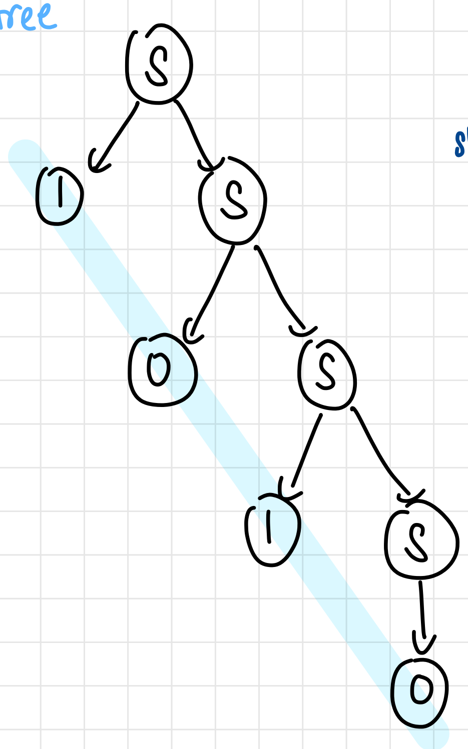
$S \overset{RM}{\Rightarrow} 1S$ $(S \rightarrow 0S)$

$\Rightarrow 10S$ $(S \rightarrow 1S)$

$\Rightarrow 101S$ $(S \rightarrow 0)$

$\Rightarrow 1010$

Parse tree



skewed to the right

yield = 1010

Question 89

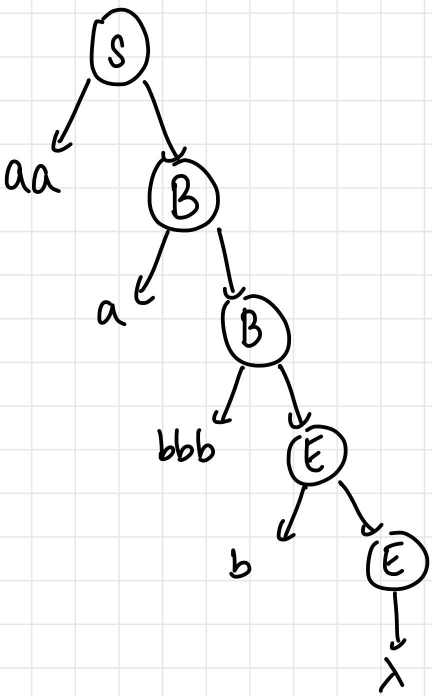
Write derivation & parse tree

$S \rightarrow aaB$
$B \rightarrow aB \mid bbbE$
$E \rightarrow bE \mid \lambda$

string = aaabbbb

$$
\begin{aligned}
S &\overset{RM}{\Rightarrow} aaB \\
&\Rightarrow aaaB \\
&\Rightarrow aaabbbE \\
&\Rightarrow aaabbbbE \\
&\Rightarrow aaabbbb
\end{aligned}
$$

Parse tree



Left Linear Grammar

Question 90

RE = a

RLG

$S \to a$

LLG

$S \to a$

Question 91

$a+b$

RRG

$S \to a \mid b$

LLG

$S \to a \mid b$

Question 92

ab

RLG

$S \to aA$
$A \to b$

LLG

$S \to Ab$
$A \to a$

Question 93

a^*

RLG

$S \to aS \mid \lambda$

LLG

$S \to Sa \mid \lambda$

Question 94

$(a+b)^*$

RLG

$S \rightarrow aS \mid bS \mid \lambda$

LLG

$S \rightarrow Sa \mid Sb \mid \lambda$

Convert RLG to LLG

- if converted to RLG by switching positions, reversal of language
- steps

finite automata for L

$\downarrow$

finite automata for L^R

$\downarrow$

RLG for L^R

$\downarrow$

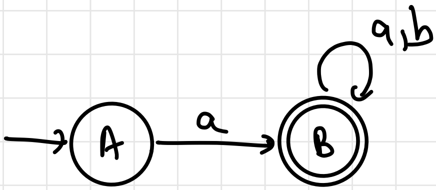
LLG for $(L^R)^R = L$

Question 95

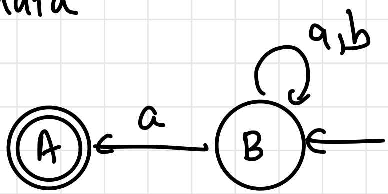
$A \to aB$
$B \to aB \mid bB \mid \lambda$

$L = \{\text{starts with } a\}$

automata



reversed automata



write RLG

$B \to aB \mid bB \mid aA$
$A \to \lambda$

convert to LLG

$B \to Ba \mid Bb \mid Aa$
$A \to \lambda$

Convert LLG to finite automata

1) swap (make RLG of L^R)

2) Make automata

3) Reverse

Question 96

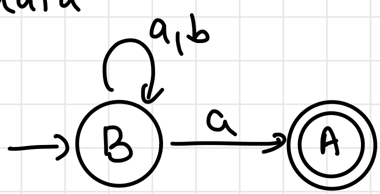
$B \rightarrow Ba \mid Bb \mid Aa$
$A \rightarrow \lambda$

convert to automata

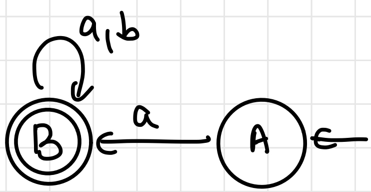
RLG of L^R

$B \rightarrow aB \mid bB \mid aA$
$A \rightarrow \lambda$

automata



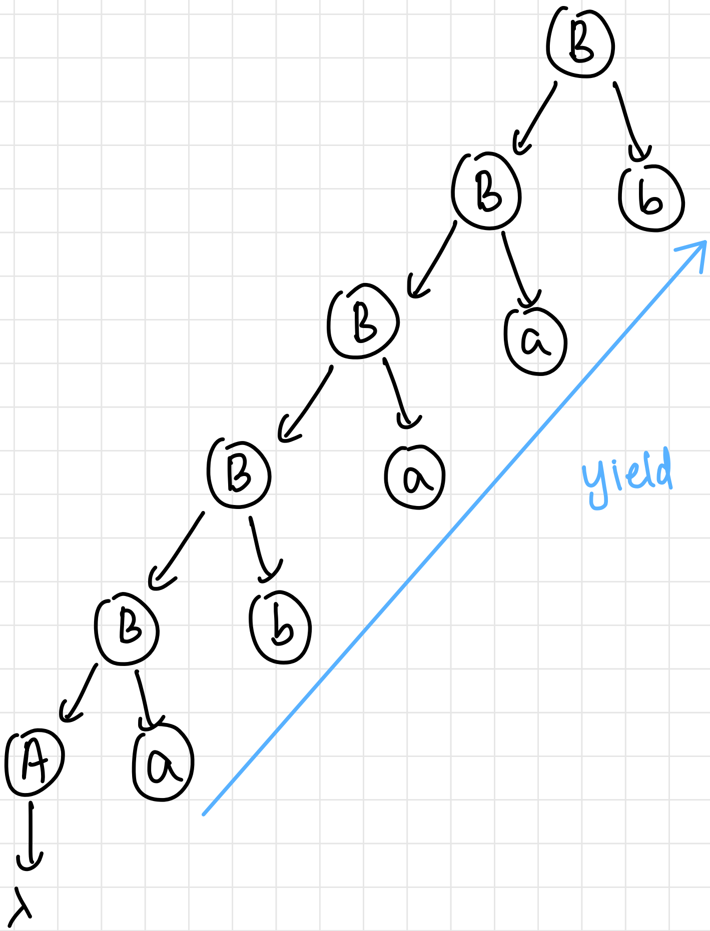
reversed



Parse tree for LLG

$B \to Ba \mid Bb \mid Aa$
$A \to \lambda$

abaab



Properties of Regular Languages

1. Concatenation
   - if L_1, L_2 are RLs, $L_1 \cdot L_2$ is RL
2. Union
   - if L_1, L_2 are RLs, $L_1 \cup L_2$ is RL
3. Closure
   - if L is RL, L^* is RL
4. Reversal
   - if L is RL, L^R is RL
5. Complement
   - if L is RL, $\bar{L}$ is RL
   - change FS → NFS & NFS → FS
   - only on DFA
6. Intersection
   - L_1 & L_2 RLs, $L_1 \cap L_2$ RL
   - Cartesian product / cross product
   - Using De Morgan's Law
   - only if accepted by both (both final states)
7. Set difference
   - strings in L_1 and not in L_2
   - cross products

Decidable Properties of Regular Languages

1) Testing emptiness

- is given RL empty
- no path from q_s to $F = \{ \dots \}$
- λ is a valid path; not empty L
- if grammar is non-producing

2) Membership

- given a string w, check if it belongs to the language or not

3) Finite / infinite

- loops: infinite

4) $L = \Sigma^*$

- accepts everything
- find complement of language and check if it accepts nothing

5) $L_1 = L_2$

- construct DFAs and minimise

Pumping Lemma

- check if language is regular or not
- when impossible to convert language to DFA/NFA: not regular
- languages are finite or infinite ($* \rightarrow$ infinite)
- finite languages are always regular
- $a^* b^*$ is infinite, regular

Question 97

Show that $L = \{a^n b^n\}$ is irregular

Pigeon hole principle

divide string into x, y & z

$n \rightarrow$ pigeons
$m \rightarrow$ pigeon holes

if no. of pigeons > no. of holes,
one hole will have > 1 pigeons

$|xy| \leq n$ $\quad |y| \geq 1$ $\quad z$

$$\begin{array}{cc} a^n & b^n \\ \downarrow & \downarrow \\ xy & z \end{array}$$

$\underbrace{aaaa}_{x}\ \underbrace{a}_{y}\ \underbrace{bbbbb}_{z}$ if $y = 2$, lang not regular

Question 98

w| equal no. of 0's and 1s

bad choice : $(01)^n$

good choice : $0^n 1^n$ → irregular

$\underbrace{00}_{xy}1\underbrace{1}_{z}$

	x	y	z
	0	0	11

$i=0$

011 ✗

$i=1$

0011 → bad eg

$i=2$

00011 ✗

Question 99

Palindrome

$a^n b^m b^m a^n$ $\qquad ww^R \mid w \in \{a,b\}^*$

Assume n states in machine

Assume ww^R is regular

1) $a^n b b a^n$

$|xy| \leq n$

$\underbrace{a^n}_{xy} \mid \underbrace{bba^n}_{z} \rightarrow xyz$

$n=3$

$\underbrace{aa}_{x}\underbrace{a}_{y} \mid bbaaa$

$xy^iz \qquad |y| \geq 1$

$i=1 \longrightarrow$ bad choice

$i=2$

$aa\,aa\,bbaaa \notin L$

Revision-Regular Expressions

Question 1

Write RE for $L = \{ w \mid w \neq 000, \{0,1\}^* \}$

$|w| = 3$, $w \neq 000$

$|w| < 3$ or $|w| > 3$

$$\lambda + 0 + 1 + (0+1)^2 +$$

$$001 + 010 + 011 + 100 + 101 + 110 + 111 +$$

$$(0+1)^4 (0+1)^*$$

Question 2

All strings w such that in every prefix of w, the no. of 0's and 1's differ by at most 1.

$$(01+10)^* (\lambda + 0 + 1)$$

Question 3

At least 2 0's and at least one 1

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