CS & IT ENGINEERING

Theory of Computation

Finite Automata :
Closure properties – part 3

Lecture No. 19

By- DEVA Sir

TOPICS TO BE COVERED

01 closure properties

02

03

04

05

Kleene Star & Kleene plus

↳ closed for regular languages

$(\text{Reg})^* \Rightarrow$ Always Regular

$(\text{Reg})^+ \Rightarrow$ Always Regular

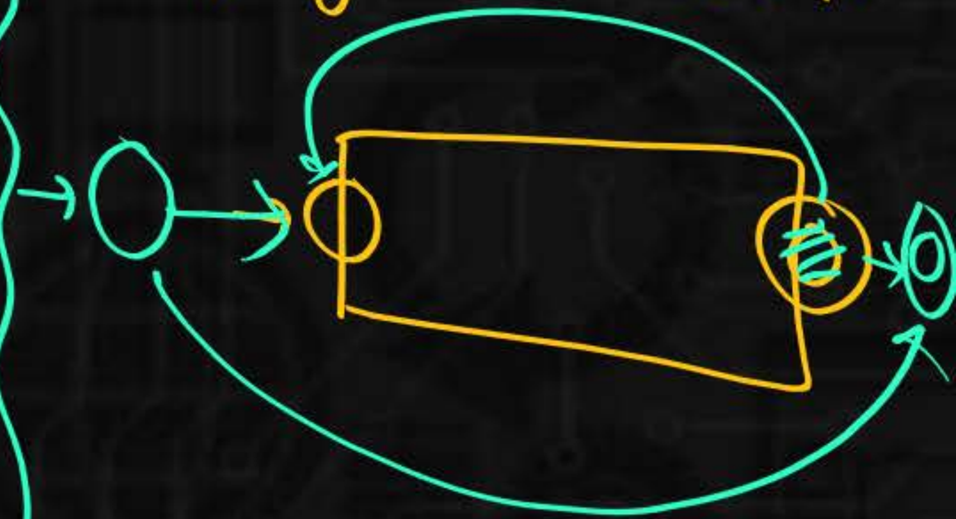
Proof 1: Use Reg Exp

$L \Rightarrow$ Reg Exp

$\Downarrow$

$L^* \Leftarrow (\text{Reg Exp})^*$

Proof 2: Use ε-NFA



$$\overline{\overline{L}} = L$$

$$\left(L^{Rev}\right)^{Rev} = L$$

$$\left(L^{*}\right)^{*}$$ = need not be L

I) If L is Reg then L^* is Regular

***II) If L^* is Reg then L need not be regular

$\left(\{a^{prime}\}\right)^*$ is Regular $\Rightarrow$ but a^{prime} is not reg

Kleene star

1. $L = a^* \Rightarrow L^* = L$

2. $L = \Sigma^* \Rightarrow L^* = L$

3. $L = \phi \Rightarrow L^* = \varepsilon$

4. $L = a^* b^* \Rightarrow L^* = (a+b)^* = (a^* b^*)^*$

5. $L = a+b \Rightarrow L^* = (a+b)^*$

6. $L = ab \Rightarrow L^* = (ab)^*$

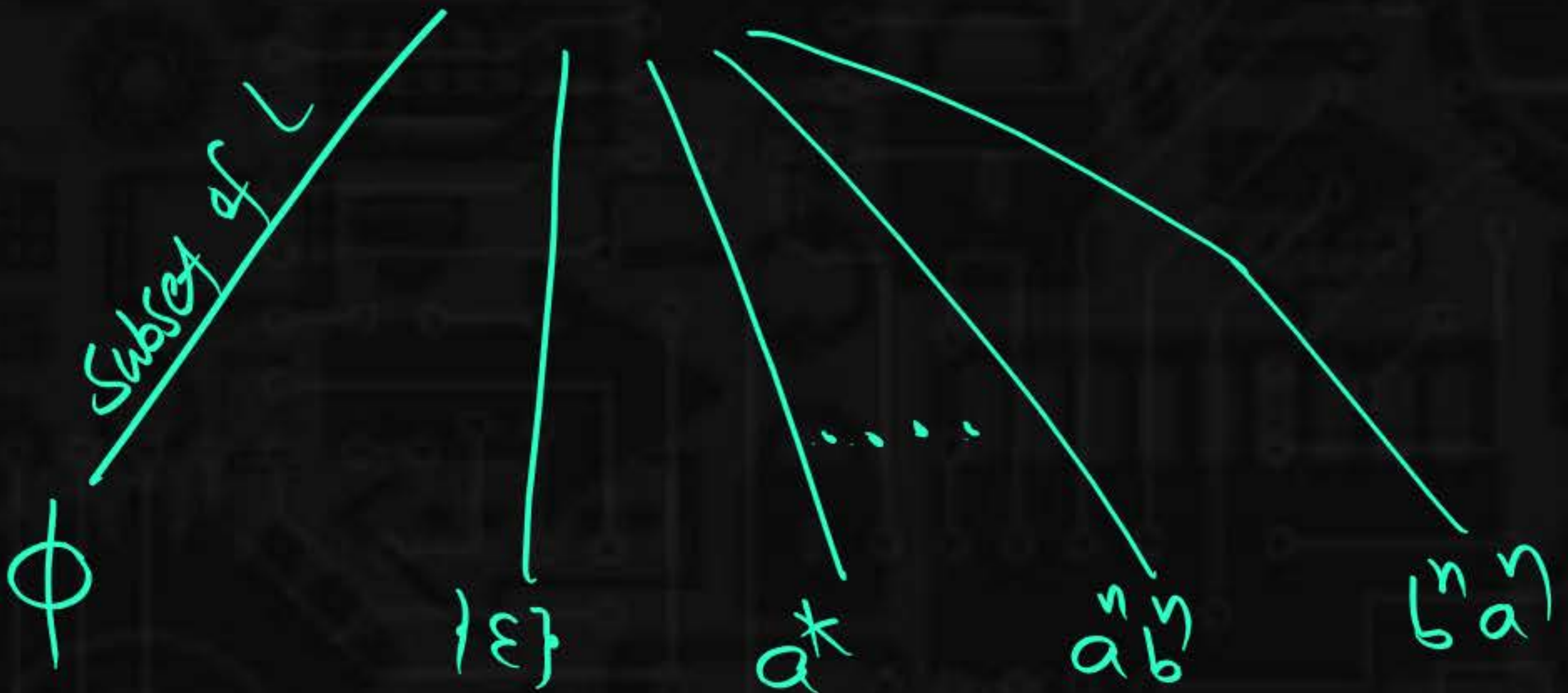
(9) Subset ($\subseteq$) ***

↳ Not closed for regular languages

Subset of regular language is <u>need not be regular</u>

L is Reg

$(a+b)^*$



Note:

Subset : $\subseteq$

proper subset : $\subset$

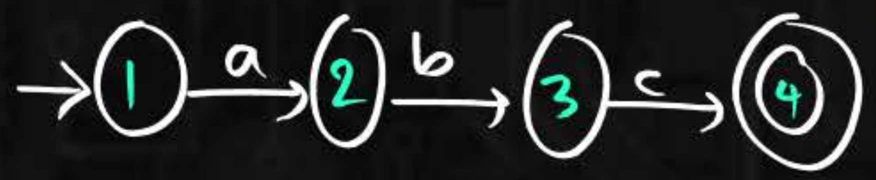
Superset : $\supseteq$

proper superset : $\supset$

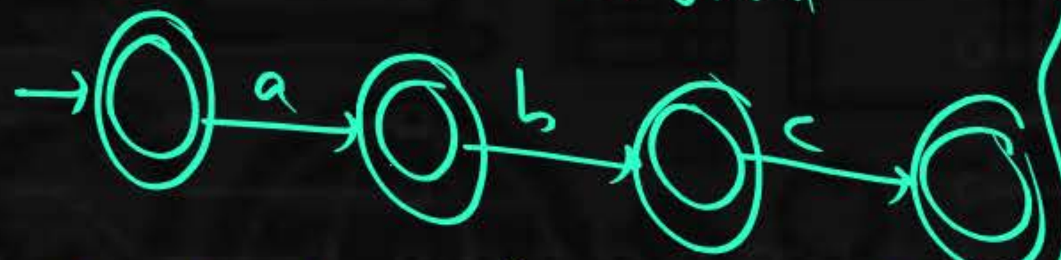
are not closed for regulars, DCFLs, CFLs, CSL, Recs, RELs

(10) Prefix(L)

$L = \{abc\}$



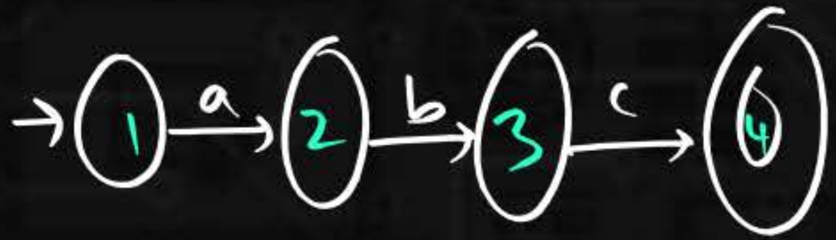
⇓ Find every path from initial to final, make every state as final



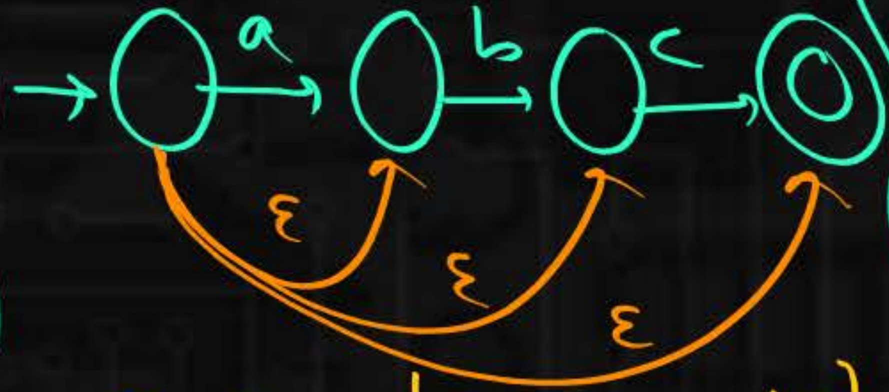
$Prefix(L) = \{\varepsilon, a, ab, abc\}$

(11) Suffix(L)

$L = \{abc\}$



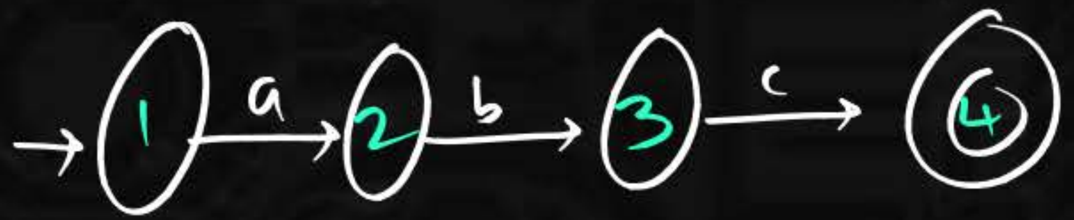
⇓



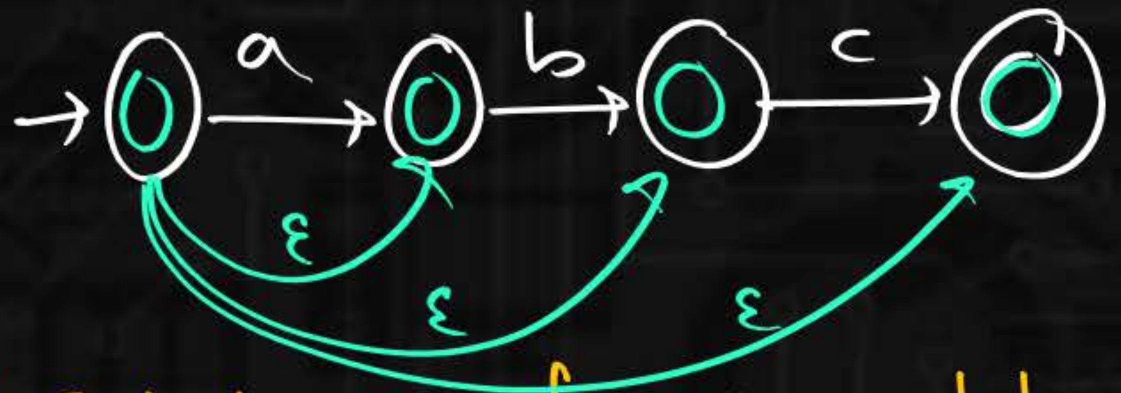
$Suffix(L) = \{\varepsilon, c, bc, abc\}$

(12) Substring(L)

$L = \{abc\}$



⇓



$Substring(L) = \{\varepsilon, a, b, c, ab, bc, abc\}$

① $L = \phi \implies$ $\text{prefix}(L) = \phi$

$\text{Suffix}(L) = \phi$

$\text{Substring}(L) = \phi$

② $L = \Sigma^* \implies$ $\text{Prefix}(L) = \Sigma^*$

$\text{Suffix}(L) = \Sigma^*$

$\text{Substring}(L) = \Sigma^*$

③ $L = \Sigma^+ \Rightarrow \text{Pref}(L) = \Sigma^*$

$\text{Suff}(L) = \Sigma^*$

$\text{Substring}(L) = \Sigma^*$

H.W.

(4) $L = a\underline{(a+b)^*}$ $\Rightarrow$ $\text{Prefix}(L) = \epsilon + L$

$\text{Suffix}(L) = (a+b)^*$

$\text{Substring}(L) = (a+b)^*$

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

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

*** (13) Quotient

$$L_1 / L_2 = \{ u \mid uv \in L_1,\ v \in L_2 \}$$

(1) $L_1 = \{ab\}$, $L_2 = \{ba\}$ $\Rightarrow$ $L_1 / L_2 = \{ab/ba\}$

$= \{\ \}$

$L_2 / L_1 = \{ba/ab\}$

$= \phi$

$\boxed{uv/v = u}$

$abcd/a = \phi$

$abcd/b = \phi$

$abcd/c = \phi$

$abc\cancel{d}/\cancel{d} = abc$

$abcd/cd = ab$

$abcd/bcd = a$

$abcd/abcd = \varepsilon$

$abcd/\varepsilon = abcd$

(2) $\left.\begin{array}{l} L_1 = a^* \\ L_2 = a \end{array}\right\} \Rightarrow L_1/L_2 = a^*/a = \{\underset{\times}{\varepsilon/a},\ \underset{\varepsilon}{a/a},\ \underset{a}{a^2/a},\ \dots\} = a^*$

$L_2/L_1 = a/a^* = \{\underset{a}{a/\varepsilon},\ \underset{\varepsilon}{a/a},\ \underbrace{a/a^2,\ \dots}_{\times}\} = \varepsilon + a$

---

(3) $\left.\begin{array}{l} L_1 = a^+ \\ L_2 = a \end{array}\right\} \Rightarrow L_1/L_2 = a^+/a = \{\underset{\varepsilon}{a/a},\ \underset{a}{aa/a},\ \underset{a^2}{aaa/a},\ \dots\} = a^*$

$L_2/L_1 = a/a^+ = \{a/a,\ \underbrace{a/a^2,\ a/a^3,\ \dots}_{\times}\} = \varepsilon$

(4) $L_1 = a^* b$

$L_2 = a b^*$

$\Rightarrow L_1 / L_2 =$

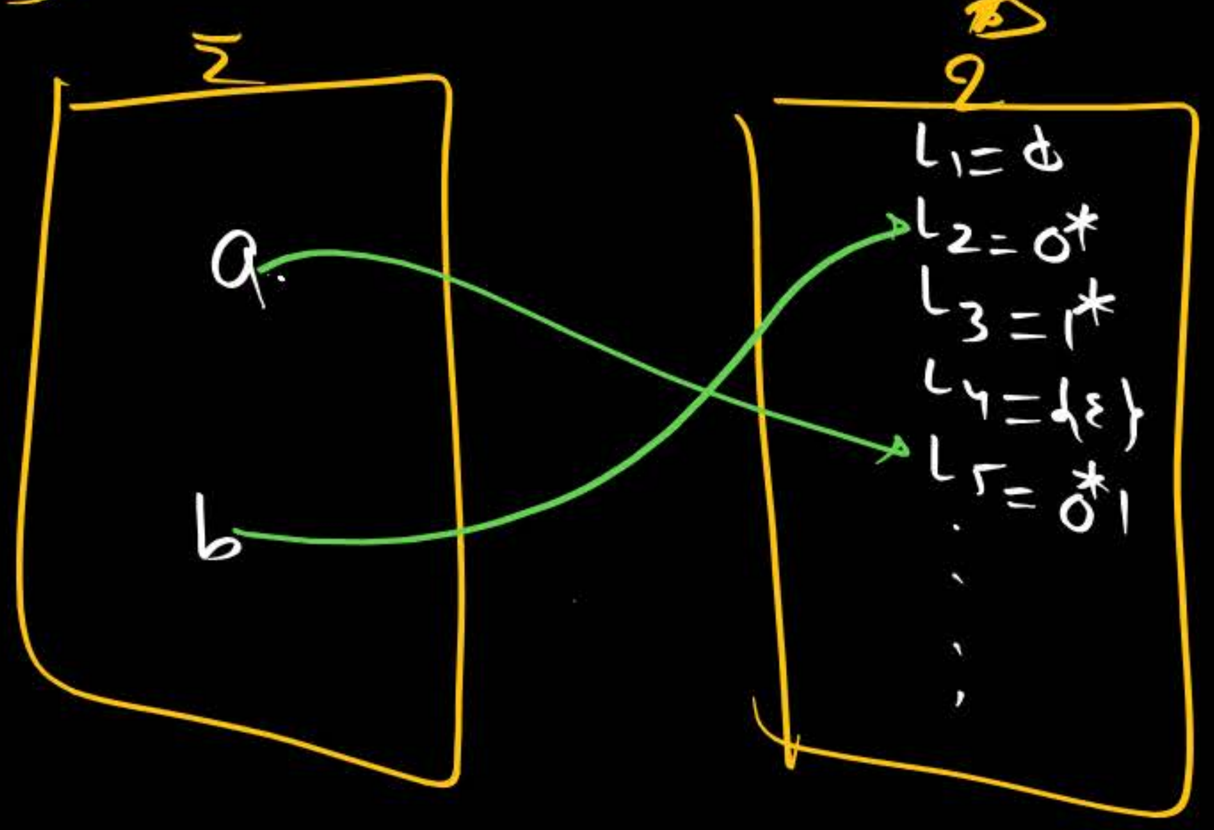
$L_2 / L_1 =$

H.W.

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

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

$$f : \Sigma \longrightarrow 2^{\Delta^*}$$



Σ = Set of all symbols

Σ^* = Set of all strings

= language of all strings

$2^{\Sigma^*} = P(\Sigma^*)$ = Set of all sets

= set of all languages

= language of all langs

(14) Substitution $f(L)$ [Regular language substitution]

$\Sigma = \{a, b\}$ $\Delta = \{0, 1\}$

$L = a^*bb$ → $f : \Sigma \to P(\Delta^*)$ → $f(L) = (0^*1)^*1111 = [f(a)]^* f(b) f(b)$

Reg Regular

Every symbol in Σ is mapped with $\boxed{\text{Some regular language}}$

$f(a)$ = Some reg
0^*1

$f(b)$ = Some reg
$\{11\}$

$f(L)$:

Given regular language L,

every symbol in Σ is substituted

with some regular language

(15) Homomorphism $h(L)$ [string substitution]

L is Reg

$L = a^*bb$

$h : \Sigma \to \Delta^*$

$h(L) = [h(a)]^* h(b) \cdot h(b)$

$= (011)^* \varepsilon\varepsilon$

$= (011)^*$

$= (011)^*$

$h(a) =$ Some string
011

$h(b) =$ Some string
ε

(16) ε-free Homomorphism [non empty string substitution]

$L = (ab)^*$ $\longrightarrow$ $\boxed{\varepsilon\text{-free } h : \Sigma \longrightarrow \Delta^+}$ $\longrightarrow$ ε-free $h(L) = (01111)^*$

$f(a) =$ non empty string
011

$f(b) =$ non empty string
11

(17) $h^{-1}(L)$

$L = \{000, 00\}$

$h(a) = 0$ $\qquad$ $h(b) = 00$

First Find h^{-1} for strings

$h^{-1}(0) = a$

$h^{-1}(00) = b$

Second: $h^{-1}(000) = \{aaa, ab, ba\}$

$h^{-1}(00) = \{aa, b\}$

$h^{-1}(L) = h^{-1}(000) \cup h^{-1}(00) = \{b, aa, ab, ba, aaa\}$

$000 \to aaa$

$000 \to ba$ $\qquad$ $000 \to ab$

(18) Half(L)

(19) Second Half(L)

(20) $\frac{1}{3}(L)$

(21) middle $\frac{1}{3}(L)$

(22) Last $\frac{1}{3}(L)$

$\varepsilon\ \varepsilon\ \varepsilon \qquad a\ b\ b$

$L = \{\varepsilon,\ a,\ ab,\ abb,\ aaba,\ ababa\}$

$|\varepsilon| = |\varepsilon| \qquad \underbrace{|a| = |b|}_{\text{2 equal lengths}} \qquad aa\ \ ba$

$\text{Half}(L) = \text{First } \frac{1}{2}(L) = \{\varepsilon, a, aa\} = \{u \mid uv \in L, |u| = |v|\}$

$\text{Second Half}(L) = \{\varepsilon, b, ba\} = \{v \mid uv \in L, |u| = |v|\}$

$\frac{1}{3}(L) = \{\varepsilon, a\} = \{u \mid uvw \in L, |u| = |v| = |w|\}$

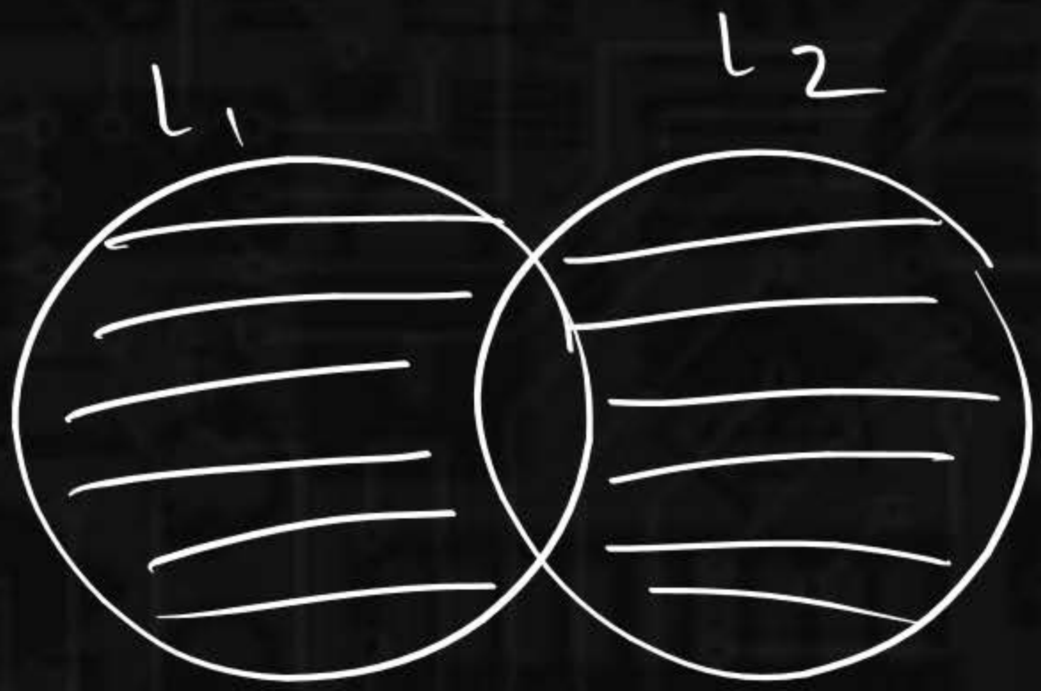
$\text{middle } \frac{1}{3}(L) = \{\varepsilon, b\} = \{v \mid \text{''} \}$

$\text{Last } \frac{1}{3}(L) = \{\varepsilon, b\} = \{w \mid \text{''} \}$

(23) Symmetric Difference (Δ)

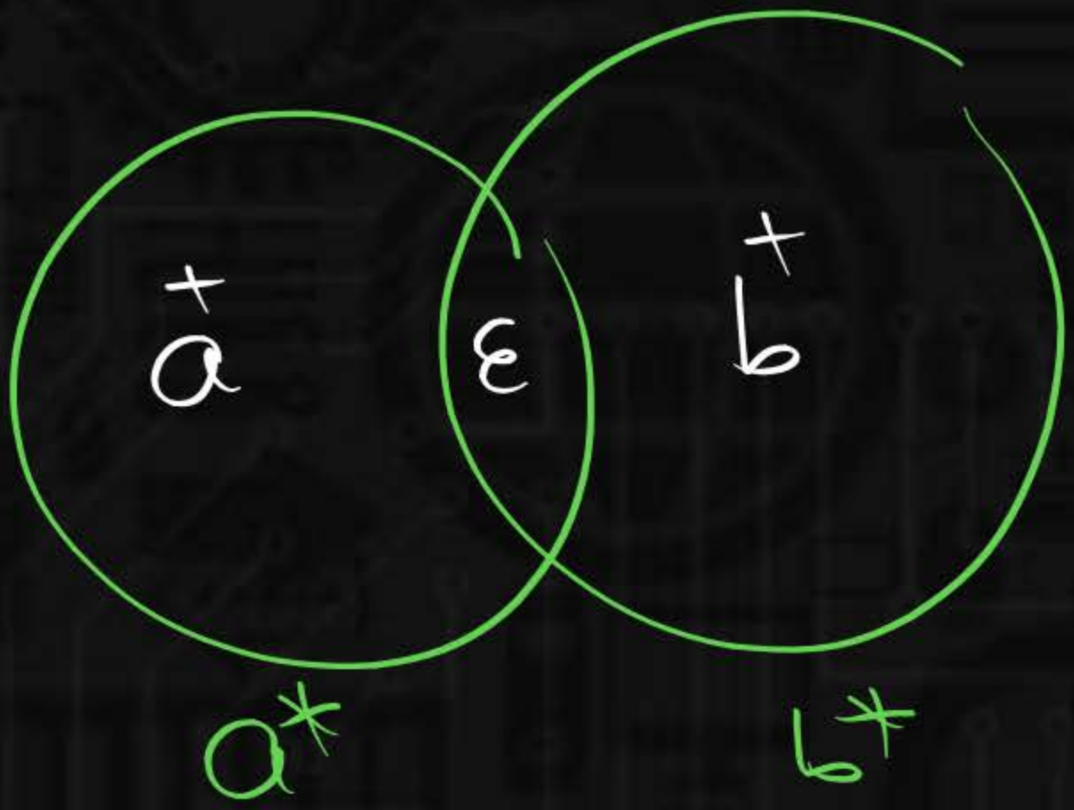
$$L_1 \Delta L_2 = (L_1 - L_2) \cup (L_2 - L_1)$$

$$= (L_1 \cup L_2) - (L_1 \cap L_2)$$



1. $\left.\begin{aligned} L_1 &= a^* \\ L_2 &= b^* \end{aligned}\right\} \Rightarrow L_1 \Delta L_2 = a^+ + b^+$

2. $\left.\begin{aligned} L_1 &= a^+ \\ L_2 &= b^+ \end{aligned}\right\} \Rightarrow L_1 \Delta L_2 = a^+ + b^+$



Note: $L_1 \circledast L_2 = \left(\overline{L_1} \cap L_2\right)^{Rev}$

If L_1 and L_2 are regular languages

then $L_1 \circledast L_2$ is Regular

$= \left(\overline{Reg} \cap Reg\right)^{Rev}$

$= \left(Reg \cap Reg\right)^{Rev}$

$= (Reg)^{Rev} \Rightarrow Reg$

(24) Finite Union: $L_1 \cup L_2 \cup \cdots \cup L_K \Rightarrow$ Regular

(25) Finite Intersection: $L_1 \cap L_2 \cap \cdots \cap L_K \Rightarrow$ Regular

(26) Finite Difference: $L_1 - L_2 - \cdots - L_K \Rightarrow$ Regular

(27) Finite Concatenation: $L_1 \cdot L_2 \cdot \cdots \cdot L_K \Rightarrow$ Regular

$L_i \rightarrow$ Regular

K is constant (fixed)

*** (28) Finite subset(L): Subset of regular language L is always **finite set**

(29) Finite Substitution(L):

L is Reg $\longrightarrow \boxed{f_{fin\,sub}} \longrightarrow f_{fin\,sub}(L) = \text{Reg}$

$f(a) = \text{fin}$ $\quad f(b) = \text{fin}$

$L = (a+b)^*$

ϕ fin $\quad a^n b^n \mid n<100$ fin $\quad a^n \mid n<10$ fin

L

finite subset of L

Always finite set

(so, regular)

Note: I) Finite subset of reg is always finite

II) Finite subset of any lang is always finite

(30) Inf union

(31) Inf Intersection

(32) Inf Difference

(33) Inf Concatenation

(34) Inf Subset

(35) Inf Substitution

Not closed for regulars
DCFLs
CFLs
CSLs
RecLs
RELs

I) $\subseteq, \subset, \supseteq, \supset$

II) $Inf(\cup, \cap, -, \cdot, \subseteq, f)$

Not closed for regulars

III) Subset & Inf($\cup, \cap, -, \cdot, \subseteq, f$) are not closed for regs / DCFLs / CFLs / CSLs / RECs / RELs

IV) $h^{-1}(L)$ & Finite subst are closed for regs / DCFLs / CFLs / CSLs / RECs / RELs

Intersection of two regulars is regular ✓

⇓

TRUE

Summary

↳ closure properties

P.L.

myhill-nerode theorem


THANK YOU GW SOLDIERS !