CS & IT ENGINEERING

Theory of Computation

Finite Automata

Lecture No. 2

By- DEVA Sir

01 Basics

02 Regular Expressions

03 Operators

04 Basic Expressions

05 Simplification of Reg Exp

Basics :

→ Symbol, Alphabet, String, Language

- Symbol → Any / Smallest / 1 length / Building Block
- Alphabet → Set of Symbols
- String → Sequence of symbols
- Language → Set of Strings

- Operations on Strings
- Chomsky Hierarchy
- Relations on Languages, Grammars, & Automata

operations on strings:

① Reversal (unary)

② Concatenation (Binary)

③ prefix

④ Suffix

⑤ Substring

} unary operations

$w = abc$

$w^R = cba$

$|w| = |w^R|$

$w_1 = ab$

$w_2 = bc$

$w_1 . w_2 = abbc$

$w_2 . w_1 = bcab$

$|w_1 w_2| = |w_2 w_1|$

Prefix

Start is fixed

$w = \overrightarrow{abcd}$

end anywhere

Prefixes of w:

ε
a
ab
abc
abcd

Suffix

end is fixed

$w = \overrightarrow{abcd}$

Begin anywhere

Suffixes of w:

ε
d
cd
bcd
abcd

Substring

$w = \overrightarrow{abcd}$

Start anywhere, end anywhere

Substrings of w:

ε
4 ← a, b, c, d
3 ← ab, bc, cd
abc, bcd → 2
abcd → 1

$\Sigma n + 1$

$1+2+3+4+1$

Prefix (ω)

$= \{u \mid uv = \omega\}$

Beginning Sequence of ω

ω = abcd

ε . abcd = ω
a . bcd = ω
ab . cd = ω
abc . d = ω

Suffix(ω)

$= \{v \mid uv = \omega\}$

Ending sequence of ω

Substring (ω)

$\{y \mid xyz = \omega\}$

ω = abcd

Part of string

ε . ε abcd = ω
ε a bcd = ω
a b cd = ω
ab c d
abc d ε
ε ab cd
bc
cd

$w = aaaa$

Lengths : 0, 1, 2, 3, 4 — Five different lengths

Prefixes of w = $\{\varepsilon, a, aa, aaa, aaaa\}$

Suffixes of w = $\{\varepsilon, a, aa, aaa, aaaa\}$

Substrings of w = $\{\varepsilon, a, aa, aaa, aaaa\}$

$w = abcd$

Prefixes of w = $\{\varepsilon, a, ab, abc, abcd\}$

Suffixes of w = $\{\varepsilon, d, cd, bcd, abcd\}$

Substrings of w = $\{\varepsilon, a, b, c, d, ab, bc, cd, abc, bcd, abcd\}$

I) How many prefixes for n-length string ?

$= n+1$

II) How many suffixes for n-length string ?

$= n+1$

III) How many substrings for n-length string ?

min $\boxed{n+1} \leq$ No. of substrings $\leq \boxed{\frac{n(n+1)}{2}+1}$ max

If all symbols are same

If all symbols in the string are distinct

IV) How many **different length** prefixes for n-length string ?

$= n+1$

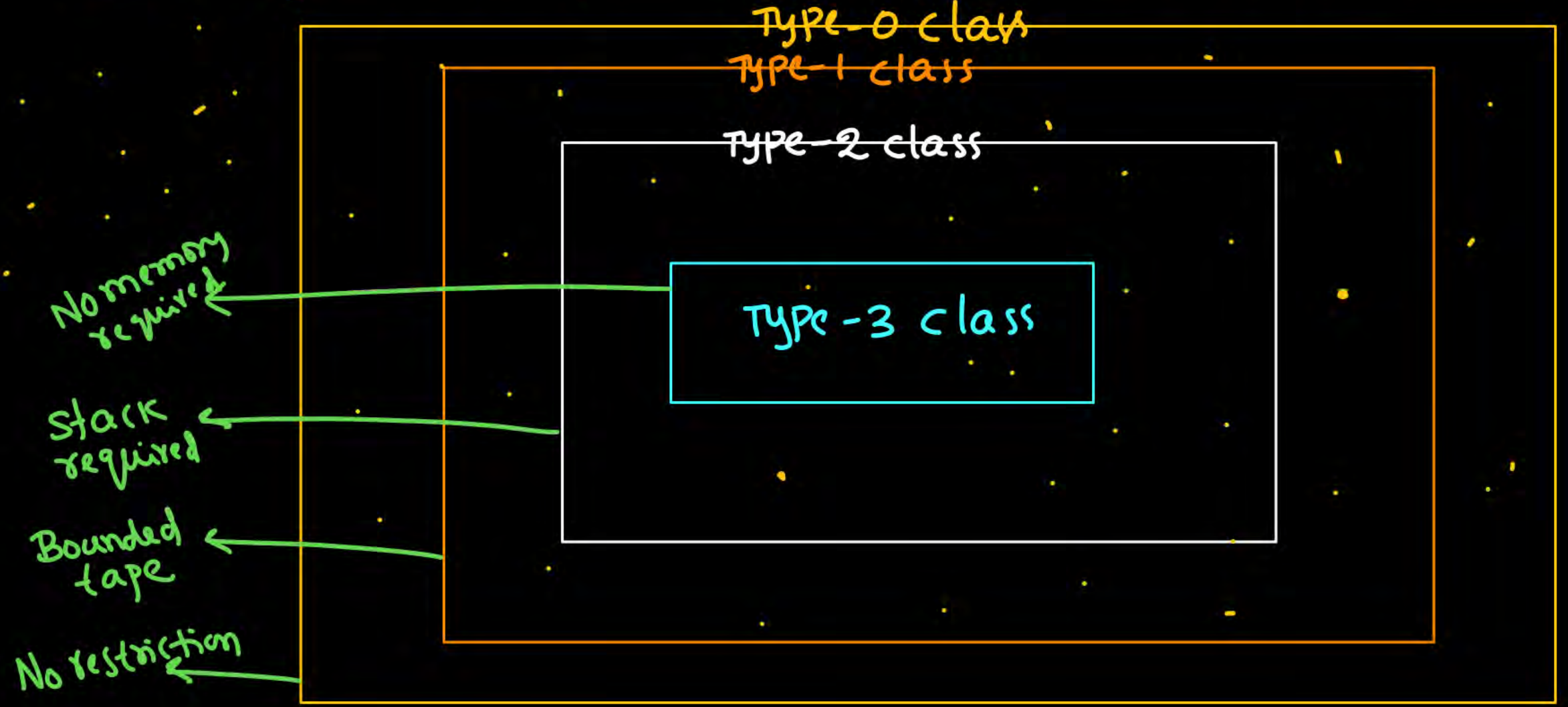
V) How many **different length** suffixes for n-length strings ?

$= n+1$

***VI) How many **different length** substrings for n-length string ?

$= n+1$

Chomsky Hierarchy : T-3 ⊂ T-2 ⊂ T-1 ⊂ T-0



Type-3 class

Set of all regular languages

Finite Automata (FA) (FSM) (FM)

Type-2 class

Set of all Context free languages

Push Down Automata (PDA)

Type-1 class

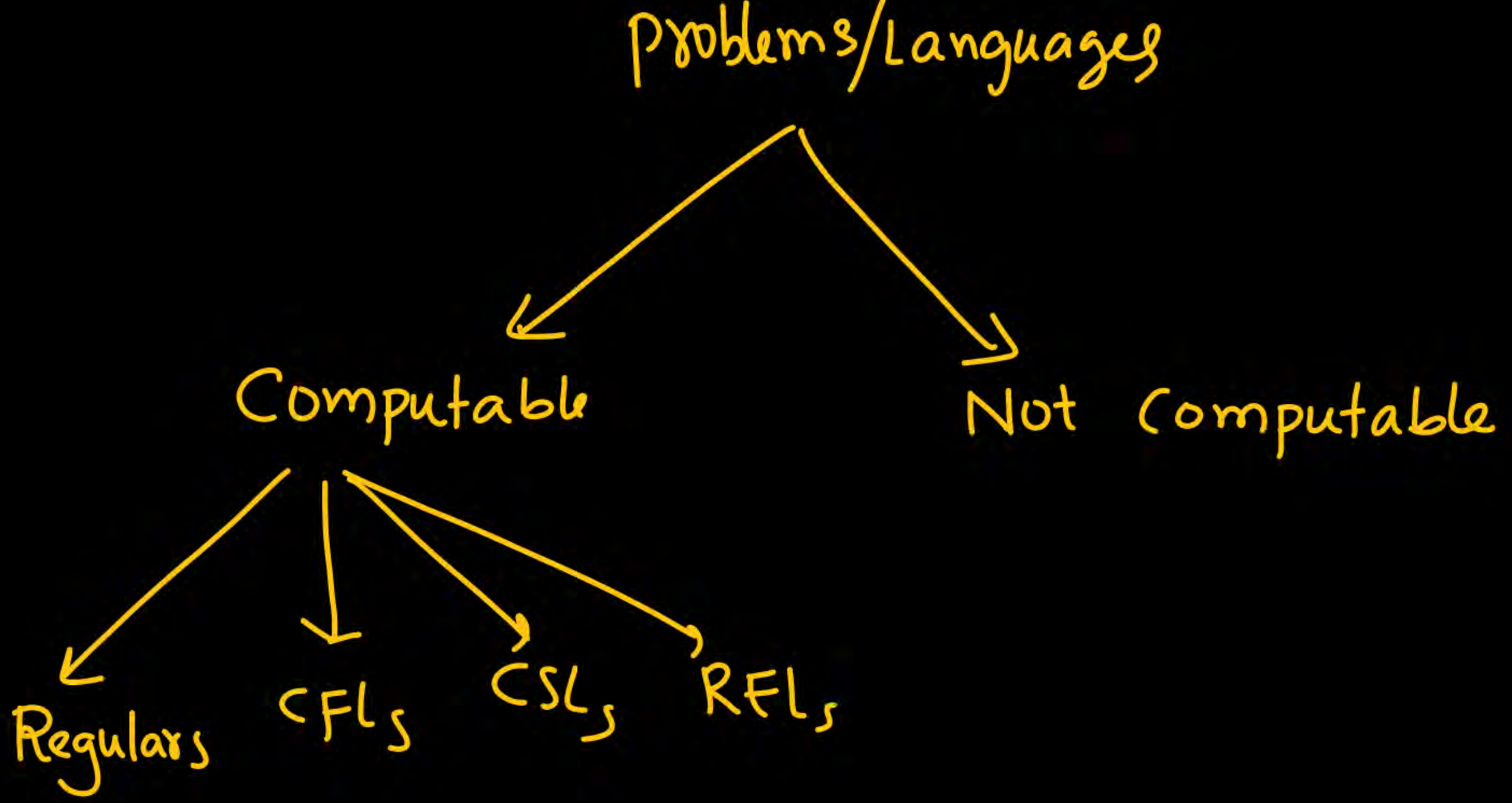
Set of all Context sensitive languages

Linear Bound Automata (LBA)

Type-0 class

Set of all Recursively Enumerable languages

Turing Machine (TM)


Problems/Languages
Computable
Not Computable
Regulars
CFLs
CSLs
RFLs

Language
↳ Set of strings

Automata (Machine)
↳ Set of states & transitions

Grammar
↳ Set of rules

Regular Language ⟺ FA ⟺ Regular Grammar

CFL ⟺ PDA ⟺ CFG

CSL ⟺ LBA ⟺ CSG

REL ⟺ TM ⟺ UG (REG)
Unrestricted Grammar

Regular Languages

- Regular Expressions
- Finite Automata (FA)
- Regular Grammar (RG)

What is the language represented by Regular Expression?
" " " " by FA ?
" " " " by RG ?

→ Reg Lang.

Problems

⇓

Languages (sets)

⇓

How to Solve ?

How to check prime?

Prime set

⇓

m/c
Grammar

Regular Expression :

- It represents (describes) (generates) a regular language.
- It uses 4 operators
  - Binary
    - I) OR (+)
    - II) Concatenation (.)
  - Unary
    - III) Kleene star [kleene closure] (*)
    - IV) Kleene plus [positive closure] (+)

OR	Concatenation	Kleene star	Kleene plus
$+, U, \mid$	$\cdot$	$*$	$+$
Binary	Binary	Unary	Unary
$R_1 + R_2$ $R_1 \cup R_2$ $R_1 \mid R_2$	$R_1 \cdot R_2$ $R_1 R_2$	R^*	R^+

OR

$R_1 + R_2$

Either R_1 or R_2

$a + b$

$\Downarrow$

$\{a, b\}$

~~aa~~ ~~ab~~ ~~ba~~

Concatenation

$R_1 \cdot R_2$

R_1 followed by R_2

$a \cdot b$

$\Downarrow$

$\{ab\}$

~~aa~~, ~~ab~~, ~~ba~~, ~~bb~~

Kleene star

R^*

Zero or more occurrences of R

$a^* = a^{\geq 0}$

$\Downarrow$

$\{a^0, a^1, a^2, a^3, \ldots\}$

$\{\varepsilon, a, aa, aaa, \ldots\}$

Kleene plus

R^+

One or more occurrences of R

$a^+ = a^{\geq 1}$

$\Downarrow$

$\{a, aa, aaa, \ldots\}$

...

$$L(a+\varepsilon) = \{\varepsilon, a\} = \{a, \varepsilon\}$$

$$L(a+ab) = \{a, ab\} = \{ab, a\}$$

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

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

ϕ : empty expression

ϕ : empty set

$L(\varepsilon) = \{\varepsilon\}$

ε ↓ non-empty expression

$\{\varepsilon\}$ → non empty set

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

$L(a) = \{a\}$

$L(a+\varepsilon) = \{a, \varepsilon\}$

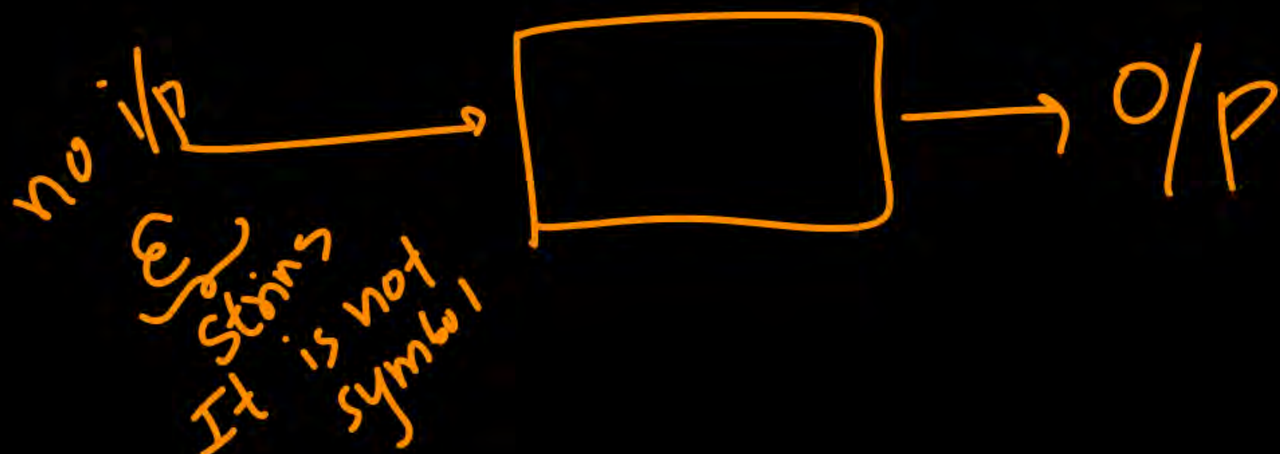
$$L(a.ab) = \{\underbrace{aab}_{\text{one string}}\}$$

$$L(a.\varepsilon) = \{a\}$$

$\varepsilon.\varepsilon = \varepsilon$	$\phi.\phi = \phi$
$a.\varepsilon = a$	$a.\phi = \phi$
$\varepsilon.a = a$	

$\underbrace{a}_{\text{one symb}}.\underbrace{\varepsilon}_{\text{zero symbol}} = a$

Why ε is important ?



OR :

1) $\varepsilon + \varepsilon = \varepsilon$

2) $\phi + \phi = \phi$

3) $a + a = a$

***4) $\varepsilon + a =$

***5) $a + \varepsilon =$

$\Rightarrow \varepsilon + a = a + \varepsilon$

6) $\varepsilon + \phi = \varepsilon$

7) $\phi + \varepsilon = \varepsilon$

8) $a + \phi = a$

9) $\phi + a = a$

Reg Exp	$\cong$	Set
ε	$\cong$	$\{\varepsilon\}$
ϕ	$\cong$	$\{\} = \phi$
a	$\cong$	$\{a\}$
$R_1 + R_2$	$\cong$	$L(R_1) \cup L(R_2)$
$R_1 \cdot R_2$	$\cong$	$L(R_1) \cdot L(R_2)$

$a + \varepsilon = a + \varepsilon$

$a \cdot \varepsilon = a$

Either a or ε

$|a| = 1$

$|\varepsilon| = 0$

$L(a+\varepsilon) = \{\varepsilon, a\}$

$a + \varepsilon \neq a$

Concatenation:

1) $\varepsilon \cdot \varepsilon = \varepsilon$

2) $\phi \cdot \phi = \phi$

3) $a \cdot a = aa = a^2$

4) $\varepsilon \cdot \phi = \phi$

5) $\phi \cdot \varepsilon = \phi$

6) $a \cdot \varepsilon = a$

7) $\varepsilon \cdot a = a$

8) $\phi \cdot a = \phi$

9) $a \cdot \phi = \phi$

$\phi \cdot R = \phi$

$R \cdot \phi = \phi$

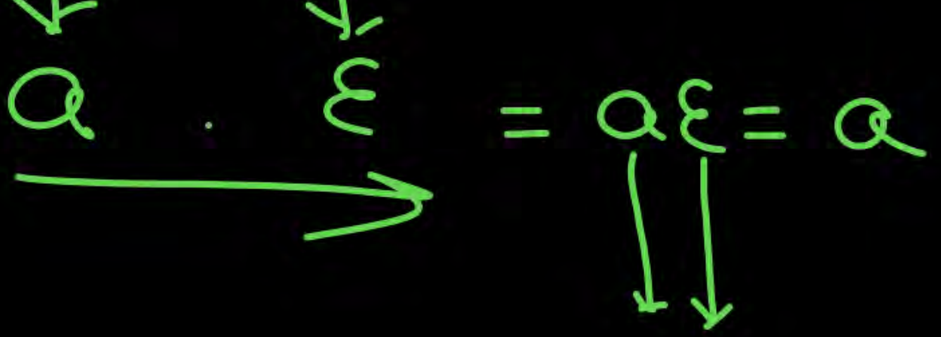
$R \cdot \varepsilon = R$

$\varepsilon \cdot R = R$

Toc: $a \cdot \varepsilon = a$

Maths/toc

$\{a\} \cdot \{\varepsilon\} = \{a\}$



ε = empty string

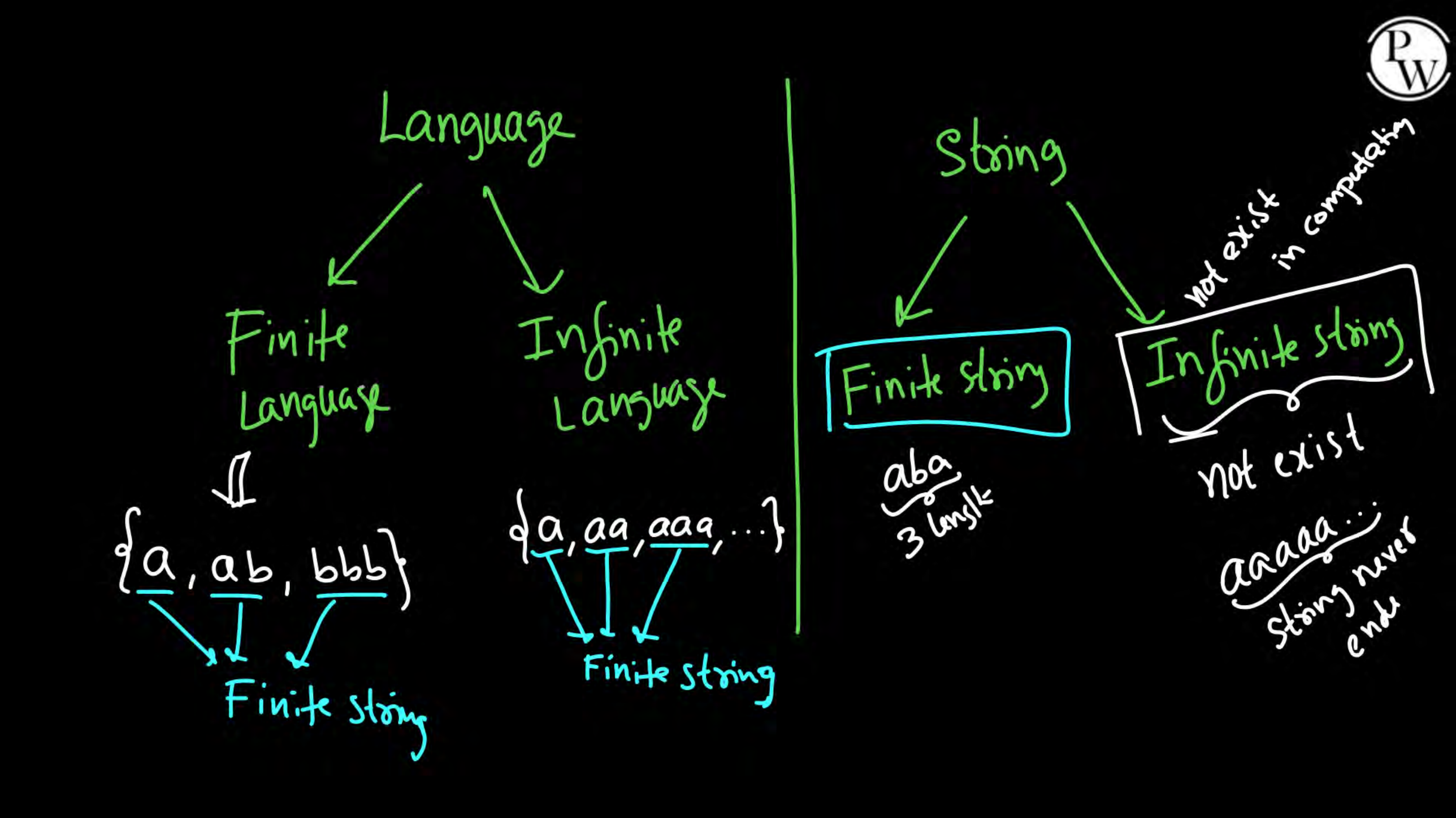
there is no symbol in sequence

$$abaaa \neq a^4b$$

$$\boxed{a^4b = aaaa.b}$$

$$ab \neq ba$$

$$aa = a^2$$


Language
Finite Language
Infinite Language
{a, ab, bbb}
Finite string
{a, aa, aaa, ...}
Finite string
String
Finite string
aba
3 length
not exist in computation
Infinite string
not exist
aaaaa...
string never ends

$(a+ab)\ (\varepsilon+aaa+b)$

$\{a, ab\} \cdot \{\varepsilon, aaa, b\} = \{a, ab, abb, a^4, abaaa\}$

$a \cdot \varepsilon \longrightarrow a$

$a \cdot aaa \longrightarrow aaaa = a^4$

$a \cdot b \longrightarrow ab$

$ab \cdot \varepsilon \longrightarrow ab$

$ab \cdot aaa \longrightarrow abaaa$

$ab \cdot b \longrightarrow abb$

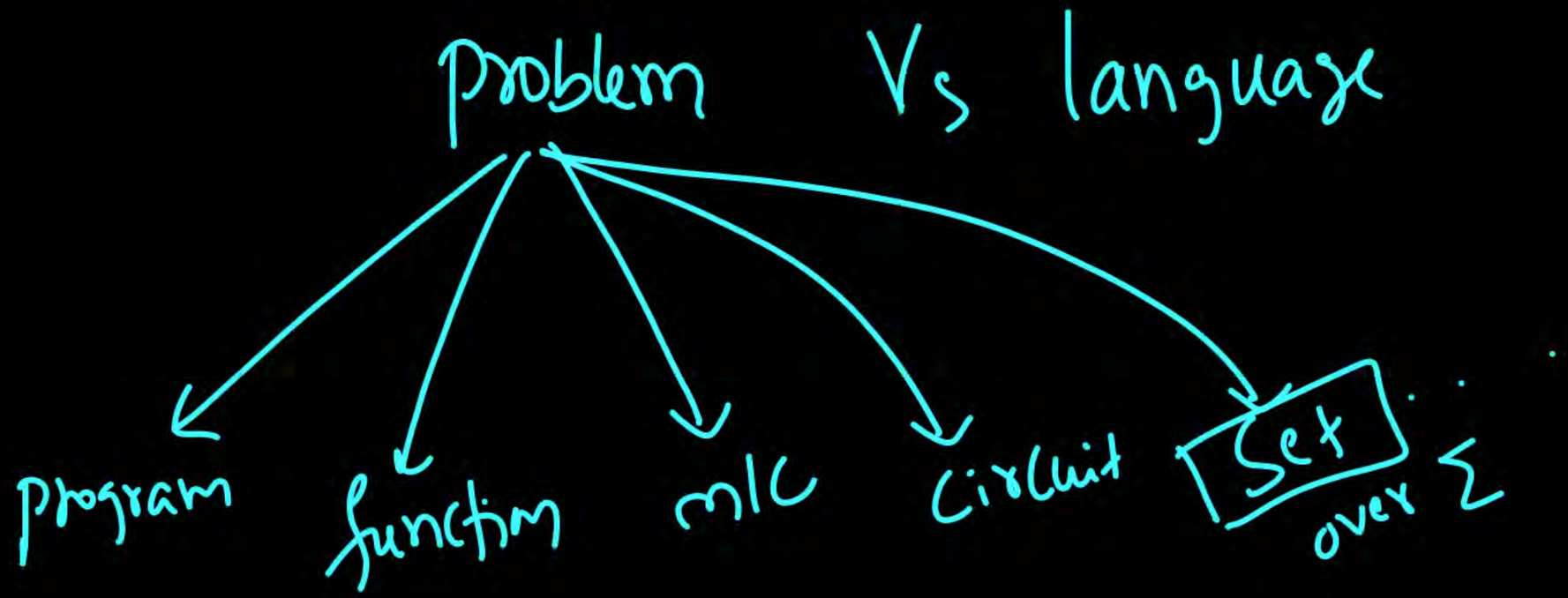
$abaaaa = ab\textcircled{a}^3 \neq (aba)^3$

$abab = (ab)^2 = (ab)\cdot(ab)$

$\underline{ab}\,\underline{ab}\,\underline{ab}\,a = (ab)^3 \cdot a$

$aaaaaa = a^5$

$(aba)^3 = \underline{aba}\,\underline{aba}\,\underline{aba}$


Problem Vs language
Program
function
m/c
circuit
Set
over Σ

Prime set $\Rightarrow$ notes: $\{2, 3, 5, 7, 11, \ldots\}$

TOC: $\{a^2, a^3, a^5, a^7, a^{11}, \ldots\}$

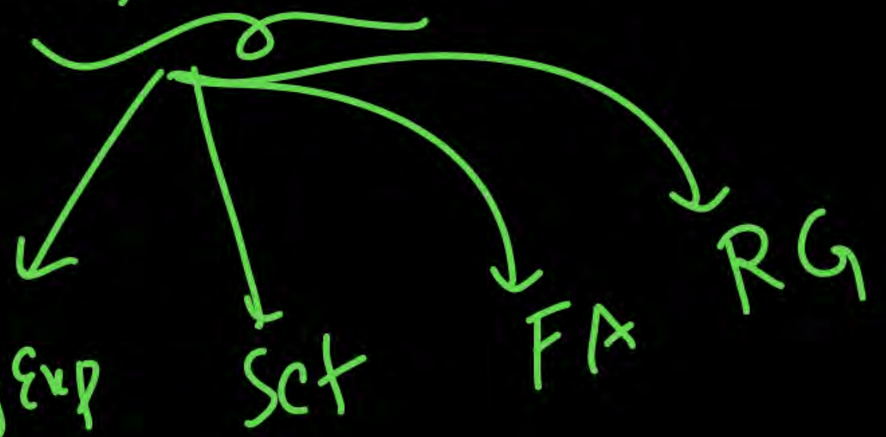
$= \{a^{\text{prime}}\}$

ε

no i/p → [ ] → produces o/p

no string
no i/p → [ ] → no o/p

Reg Exp $\cong$ Reg lang



$\varepsilon . a = a$

$\{\varepsilon\} . \{a\} = \{\varepsilon . a\} = \{a\}$

$\phi . a \neq a$

$\{\} . \{a\} = \{\}$

there is no string

Summary

- Basics
- RegExp → + , . , * , + } quantifiers ⇒ next

String

Symbol . Symbol . Symbol

a . a . b ⇒ aab

Sequence of symbols

{ String, String }

{ ε, ab }

Set


THANK YOU
GW
SOLDIERS !