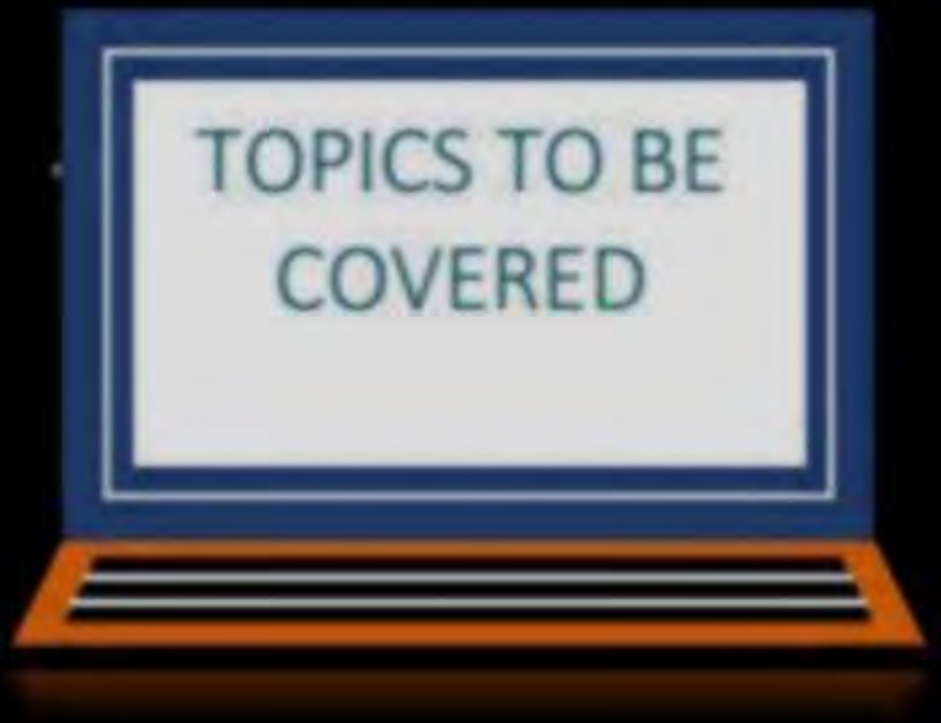
CS & IT ENGINEERING

Theory of Computation

Decidability

Lecture No. 3

By- DEVA Sir



Simplified CFG

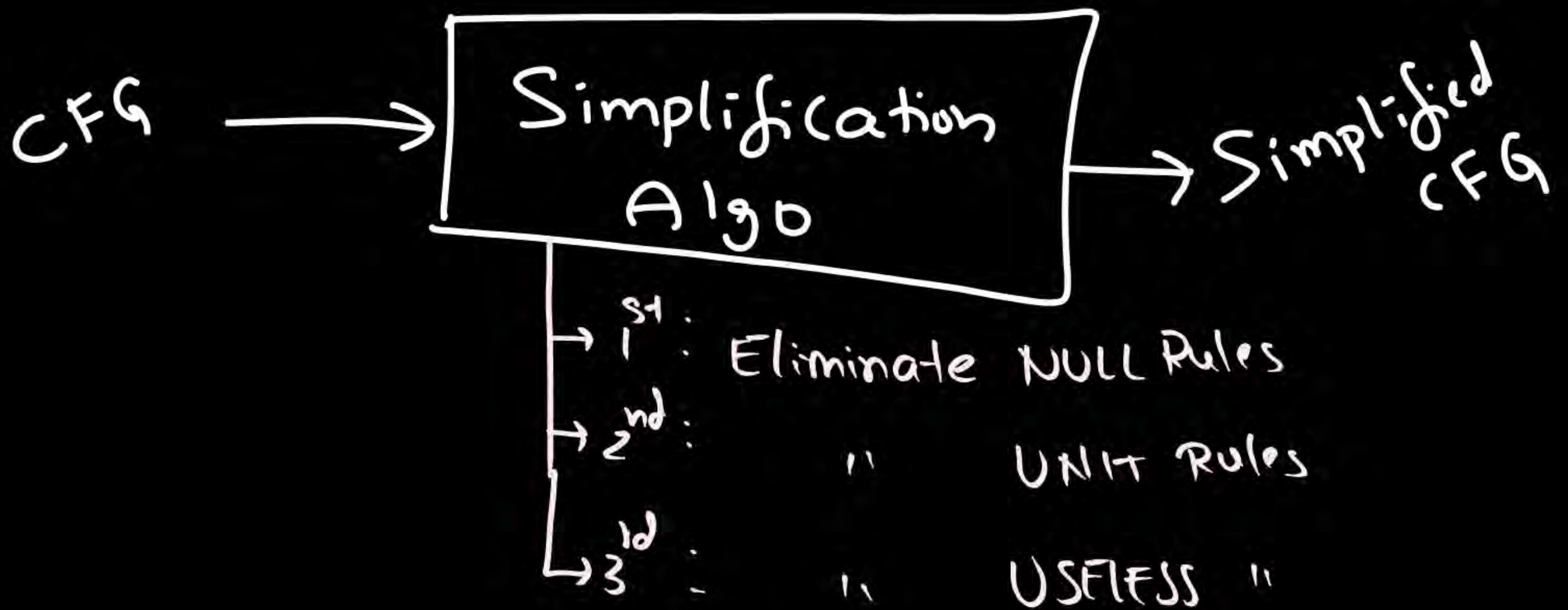
Types of TM

Decidability Questions

Complexity Theory

Revision

Simplification of CFG:



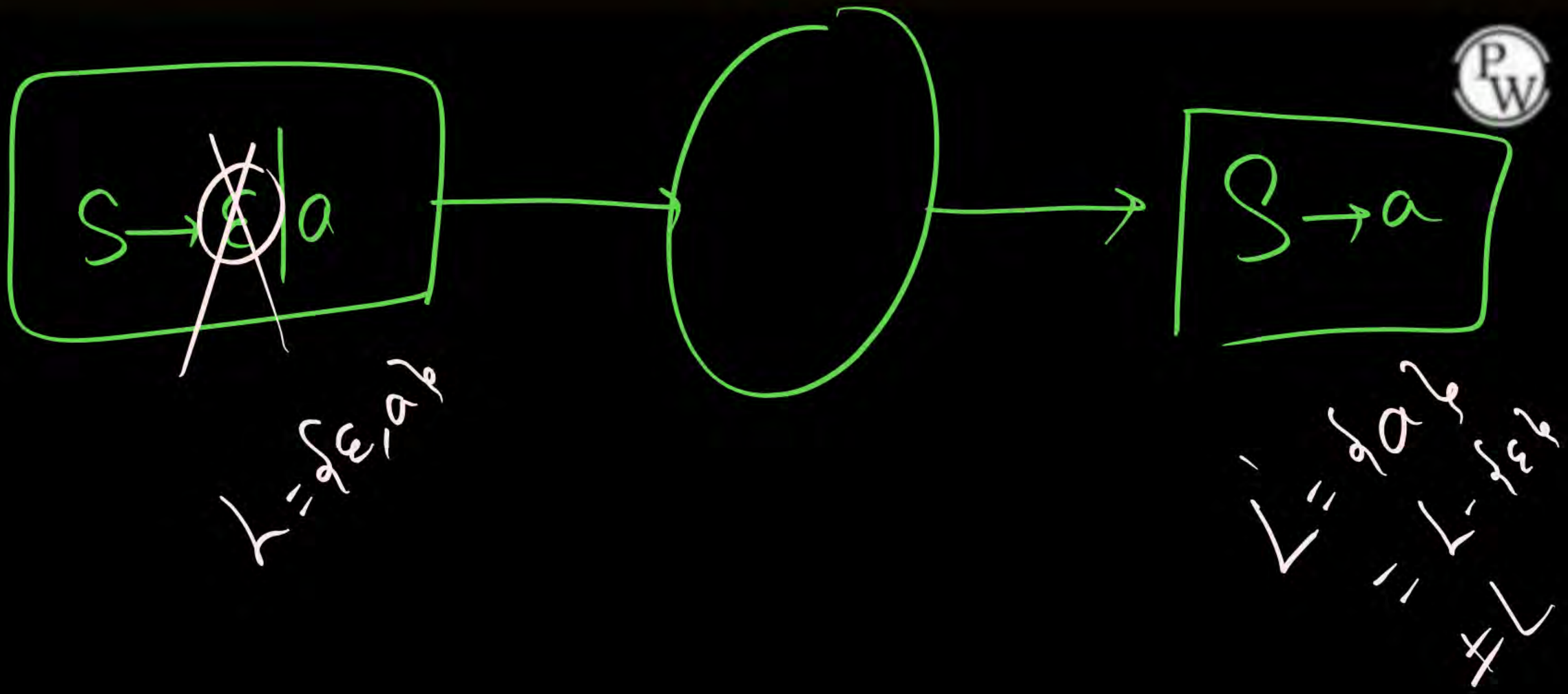
Redundant CFG

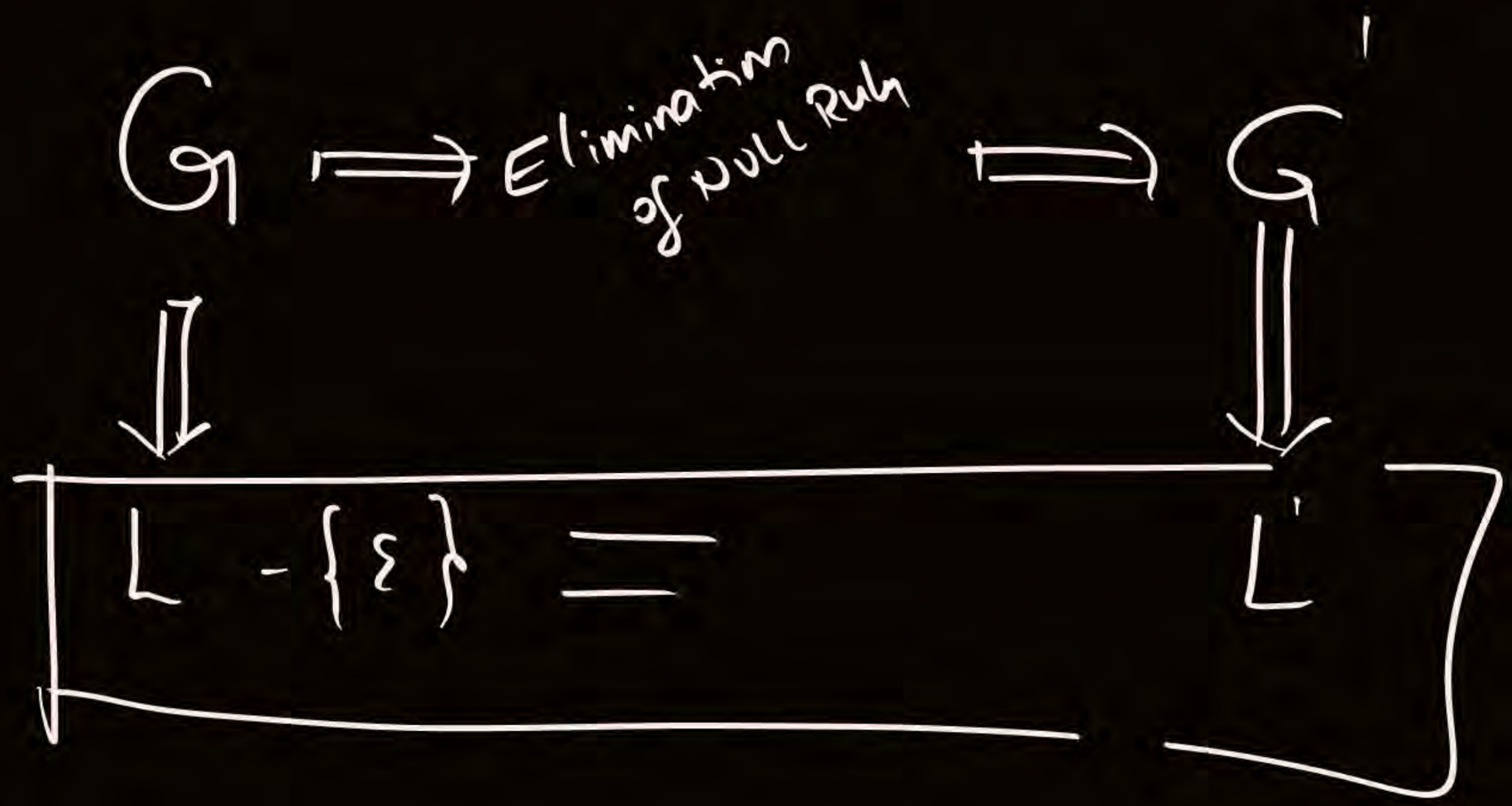
How to eliminate NULL productions ?

L

$L = L - \{\varepsilon\}$

$= L$


S → ε | a
S → a


G
Elimination of Null Rule
G'
L - {ε} =
L'

$S \rightarrow AB \mid \varepsilon$

$A \rightarrow aA \mid bB \mid c$

$B \rightarrow Ba \mid \varepsilon \mid bS$

Step 1 : $S \rightarrow \varepsilon$

$S \rightarrow AB$

$A \rightarrow aA \mid bB \mid c$

$B \rightarrow Ba \mid \varepsilon \mid bS \mid b$

Step 2 : $B \rightarrow \varepsilon$

$S \rightarrow AB \mid A$

$A \rightarrow aA \mid bB \mid c \mid b$

$B \rightarrow Ba \mid bS \mid b \mid a$

$S \to Sa \mid Ab \mid Bc \mid AB$

$A \to Ae \mid \varepsilon$

$B \to Bb \mid \varepsilon$

step 1: $A \to \varepsilon$ delete

$S \to Sa \mid Ab \mid Bc \mid AB \mid b \mid B$

$A \to Ae \mid e$

$B \to Bb \mid \varepsilon$

Step 2: Delet $B \to \varepsilon$

$S \to Sa \mid Ab \mid Bc \mid AB \mid b \mid B \mid c \mid A \mid \varepsilon$

$A \to Ae \mid e$

$B \to Bb \mid b$

Step 3: Delet $S \to \varepsilon$

$S \to Sa \mid Ab \mid Bc \mid AB \mid b \mid B \mid c \mid A \mid a$

$A \to Ae \mid e$

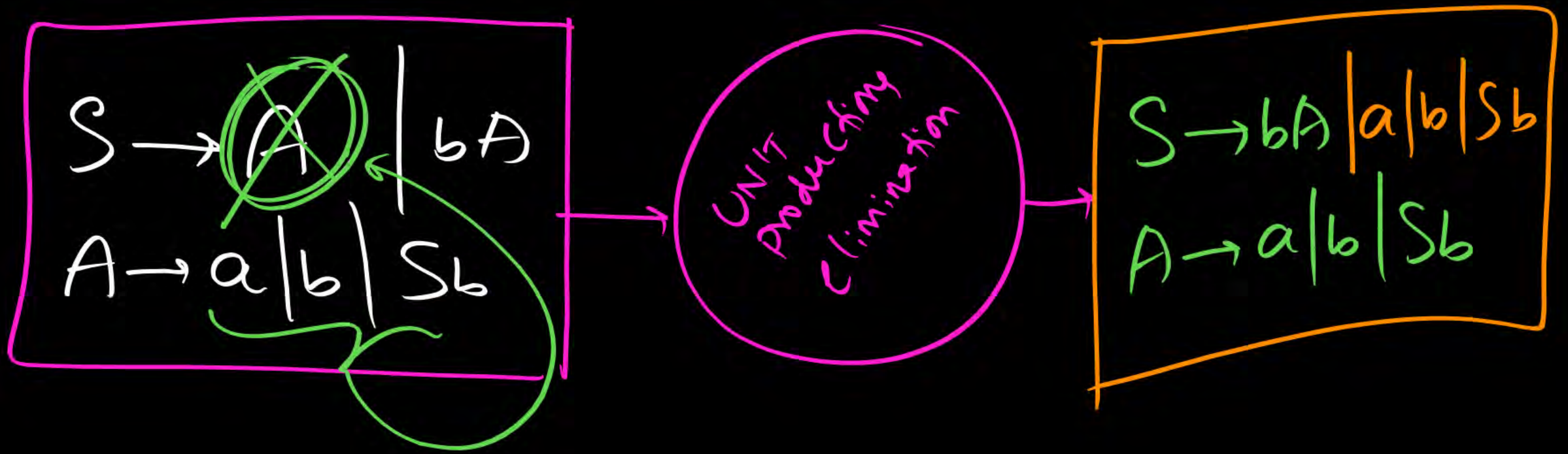
$B \to Bb \mid b$

$X \rightarrow \varepsilon$ delete

In whole grammar,

add all new productions by putting X with ε

UNIT productions Elimination :



$S \to aA \mid A$

$A \to bB \mid aS \mid B$

$B \to aA \mid S$

Step1: Delete $S \to A$

$S \to aA \mid bB \mid aS \mid B$

$A \to bB \mid aS \mid B$

$B \to aA \mid S$

Step2: Delete $S \to B$

$S \to aA \mid bB \mid aS \mid \cancel{aA} \mid \cancel{S}$

$A \to bB \mid aS \mid B$

$B \to aA \mid S$

Step3: Delete $A \to B$

$S \to aA \mid bB \mid aS$

$A \to bB \mid aS \mid aA \mid S$

$B \to aA \mid S$

Step4: Delete $A \to S$

$S \to aA \mid bB \mid aS$

$A \to bB \mid aS \mid aA$

$B \to aA \mid S$

$\Rightarrow$

Step5:

$S \to aA \mid bB \mid aS$

$A \to bB \mid aS \mid aA$

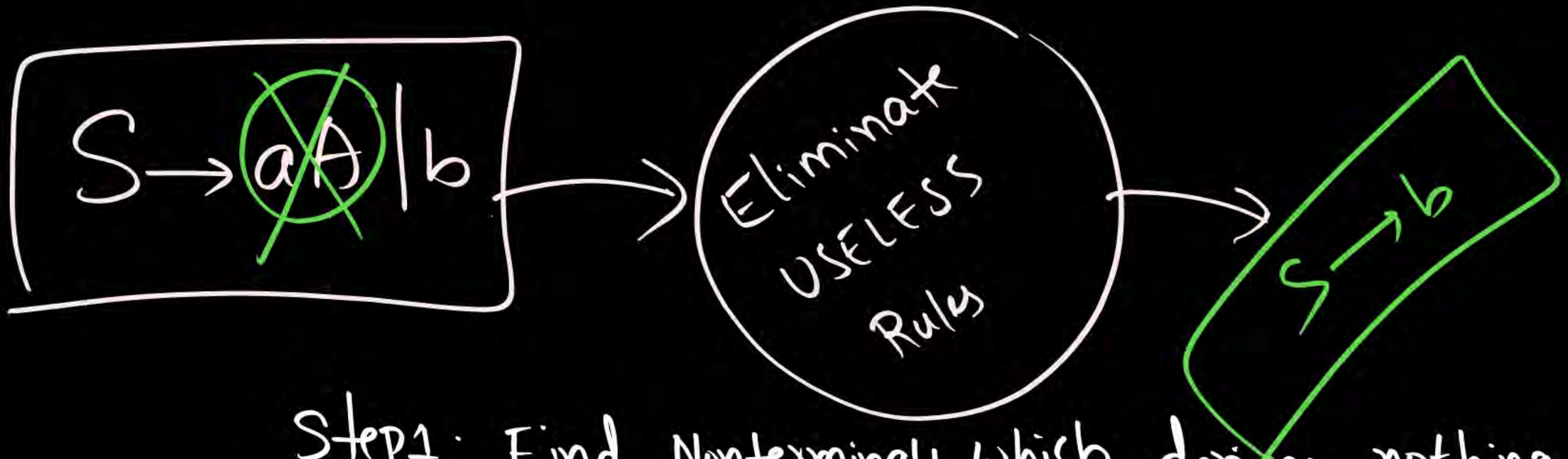
$B \to aA \mid bB \mid aS$

In given G, if we delete $X \rightarrow Y$

⇓

Replace Y with all Y productions

Eliminate of USELESS productions:



Step1: Find Nonterminals which derives nothing and eliminate it.

Step2: Find unreachable nonterminals and delete them.

$S \to aA \mid Bb \mid cDe$

$A \to a \mid Sa$

$B \to bB$

$D \to bD \mid eE$

$F \to g$

$H \to eA$

Step 1: Identify non terminals derives nothing.

S	A	B	D	E	F	H
⇓	⇓	⇓	⇓	⇓	⇓	⇓
aa	a	X	X	X	g	eA

Delete all productions associated with B, D, E.

$S \to aA$

$A \to a \mid Sa$

~~$F \to g$~~

~~$H \to eA$~~

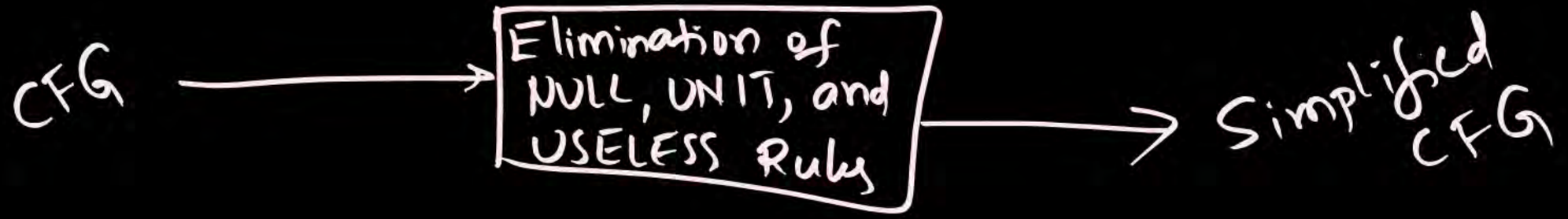
Step 2: Delete all unreachable Symbols

S → A

F and H are not reachable from S

$S \to aA$

$A \to a \mid Sa$


CFG
Elimination of NULL, UNIT, and USELESS Rules
Simplified CFG


CFG
Elimination of USELESS Rules
Reduced CFG

Normal Forms :

- CNF CFG [Chomsky Normal Form]
- GNF CFG [Greibach Normal Form]

CNF CFG

$V \rightarrow VV \mid T$

exactly 2 Non-terminal

only one terminal

Example :

$S \rightarrow SS \mid AS \mid a$

$A \rightarrow b \mid SA$

GNF CFG

$V \rightarrow TV^*$

Example :

$S \rightarrow a \mid aSSSA$

$A \rightarrow bA \mid cAAASA$

① $S \rightarrow a$

CNF ✓
GNF ✓

② $S \rightarrow ab$

CNF ✗
GNF ✗

③ $S \rightarrow SS \mid a$

CNF ✓
GNF ✗

④ $S \rightarrow aS \mid bSS \mid c$

CNF ✗
GNF ✓

How to convert CFG to CNF CFG ?

① $S \rightarrow ab$

$S \rightarrow C_1 C_2$

$C_1 \rightarrow a$

$C_2 \rightarrow b$

OR

$S \rightarrow AB$

$A \rightarrow a$

$B \rightarrow b$

(2) $S \to abc$ $\Rightarrow$ $S \to C_1C_2C_3$

$C_1 \to a$
$C_2 \to b$
$C_3 \to c$

$\Downarrow$

$S \to C_1X$
$X \to C_2C_3$
$C_1 \to a$
$C_2 \to b$
$C_3 \to c$

CNF

③ $S \to aS \mid bcA \mid bB$

$A \to abc \mid \epsilon$

$B \to Aa \mid b$

$S \to aS$

$\Downarrow$

$S \to C_a S$ ✓

$S \to bcA$

$\Downarrow$

$S \to C_2 C_3 A$

$\Downarrow$

$S \to C_2 X$

$X \to C_3 A$

$S \to bB$

$\Downarrow$

$S \to C_2 B$

$A \to abc$

$\Downarrow$

$A \to C_1 C_2 C_3$

$\Downarrow$

$A \to C_1 Y$

$Y \to C_2 C_3$

$S \to C_1 S \mid C_2 X \mid C_2 B$

$X \to C_3 A$

$A \to C_1 Y \mid \epsilon$

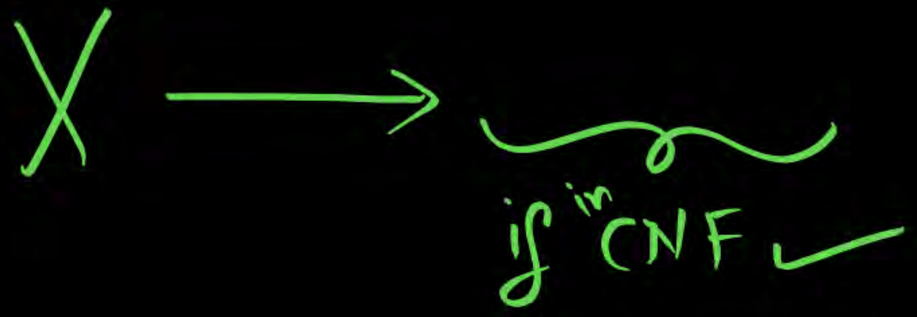
$Y \to C_2 C_3$

$B \to AC_1 \mid b$

$C_1 \to a$

$C_2 \to b$

$C_3 \to c$



if not in CNF $\Rightarrow$ make every symbol as nonterm

then divide into 2 length

CFG $\Rightarrow$ Simplified $\Rightarrow$ CNF CFG

if terminal ⇒ make nonterminal

if Nonterminal, Substitute with production

How to Convert CFG to GNF CFG ?

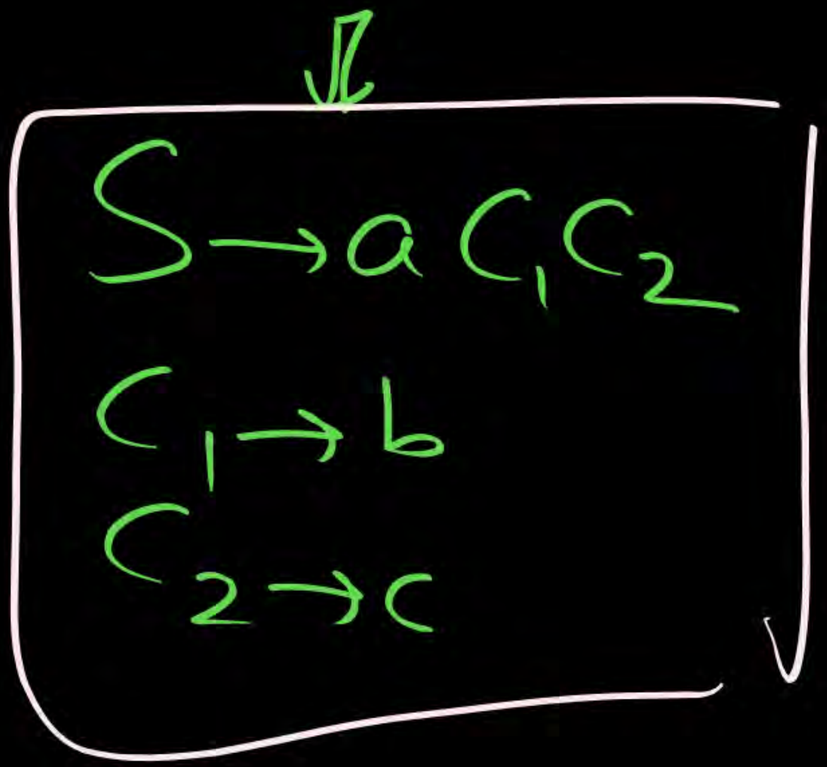
CFG $\Rightarrow$ Simplified CFG $\Rightarrow$ Non left Rec CFG $\Rightarrow$ GNF CFG

① $S \rightarrow ab$ $\Rightarrow$

$S \rightarrow aC_1$

$C_1 \rightarrow b$

(2) $S \rightarrow abc$



GNF CFG

(3) $S \rightarrow Aab \mid baB$

$A \rightarrow f$

$B \rightarrow gh$

$S \rightarrow fC_1C_2 \mid bC_1B$

$A \rightarrow f$ (معاد)

$B \rightarrow gC_3$

$C_1 \rightarrow a$

$C_2 \rightarrow b$

$C_3 \rightarrow h$

$S \rightarrow Aab$ (Sub) $\Rightarrow S \rightarrow fab \Rightarrow S \rightarrow fC_1C_2$

$S \rightarrow baB \Rightarrow S \rightarrow bC_1B$

$B \rightarrow gh \Rightarrow B \rightarrow gC_3$

How long (How many steps) the derivation takes place for deriving n length string using CNF CFG ?

i) $n = 1$ (1 length string)

$w = a$

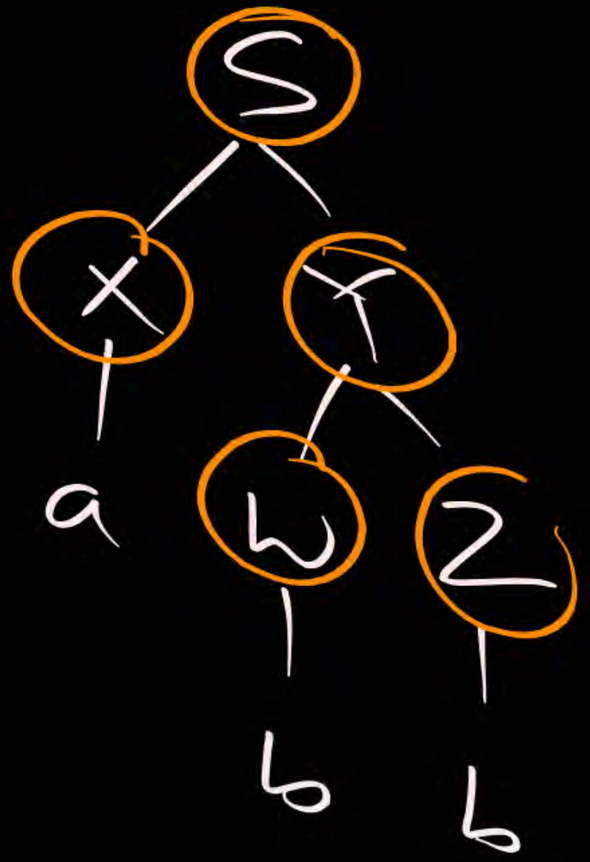
$\Rightarrow$ 1 step

ii) $n = 2$

$w = aa$

$\Rightarrow$ 3 steps

iii) $n=3$ $w=abb$



$S \Rightarrow XY$

$\Rightarrow aY$

$\Rightarrow aWZ$

$\Rightarrow abZ$

$\Rightarrow abb$

$\therefore$ 5 steps

$\Rightarrow$ 5 nonleaf nodes

$\Rightarrow$ 5 Substitutions

n length string ⇒ (2n-1) steps

taken using CNF CFG

How long (How many steps) the derivation takes place for deriving n length string using GNF CFG ?

i) $n = 1 \Rightarrow$ #steps = 1

ii) $n = 2 \Rightarrow$ #steps = 2

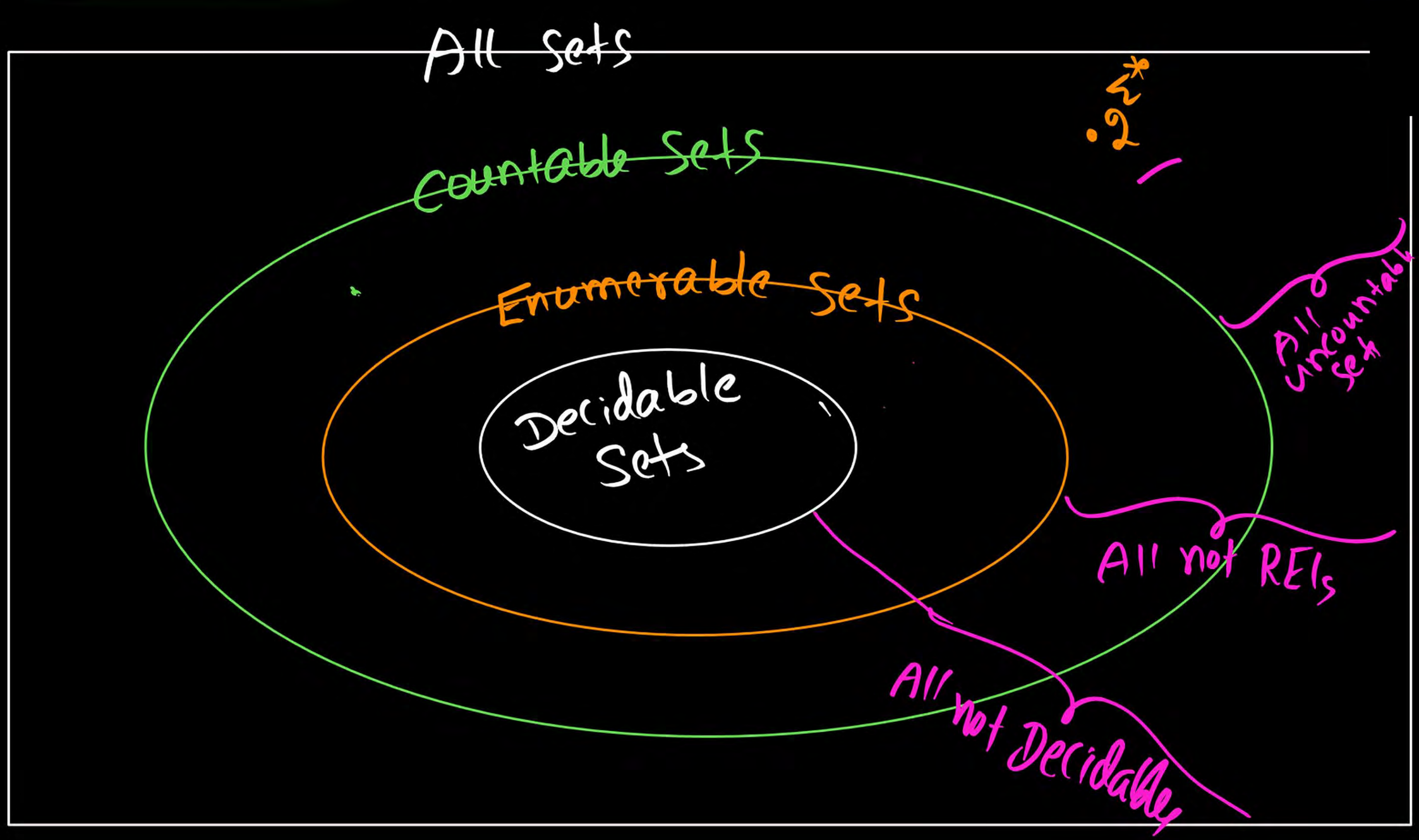
iii) $n = 3 \Rightarrow$ # steps = 3

For n length string

$\Downarrow$

n steps using GNF CFG

$n = 1$	$n = 2$	S
S	S	$\Downarrow$
$\Downarrow$	$\Downarrow$	aXY
a	aX	$\Downarrow$
	$\Downarrow$	aaY
	aa	$\Downarrow$
		aaa
$w = a$	$w = aa$	$w = aaa$


All Sets
2^{Σ^*}
Countable Sets
Enumerable Sets
Decidable Sets
All uncountable sets
All not REIs
All not Decidable

I) Every Decidable set is Countable set

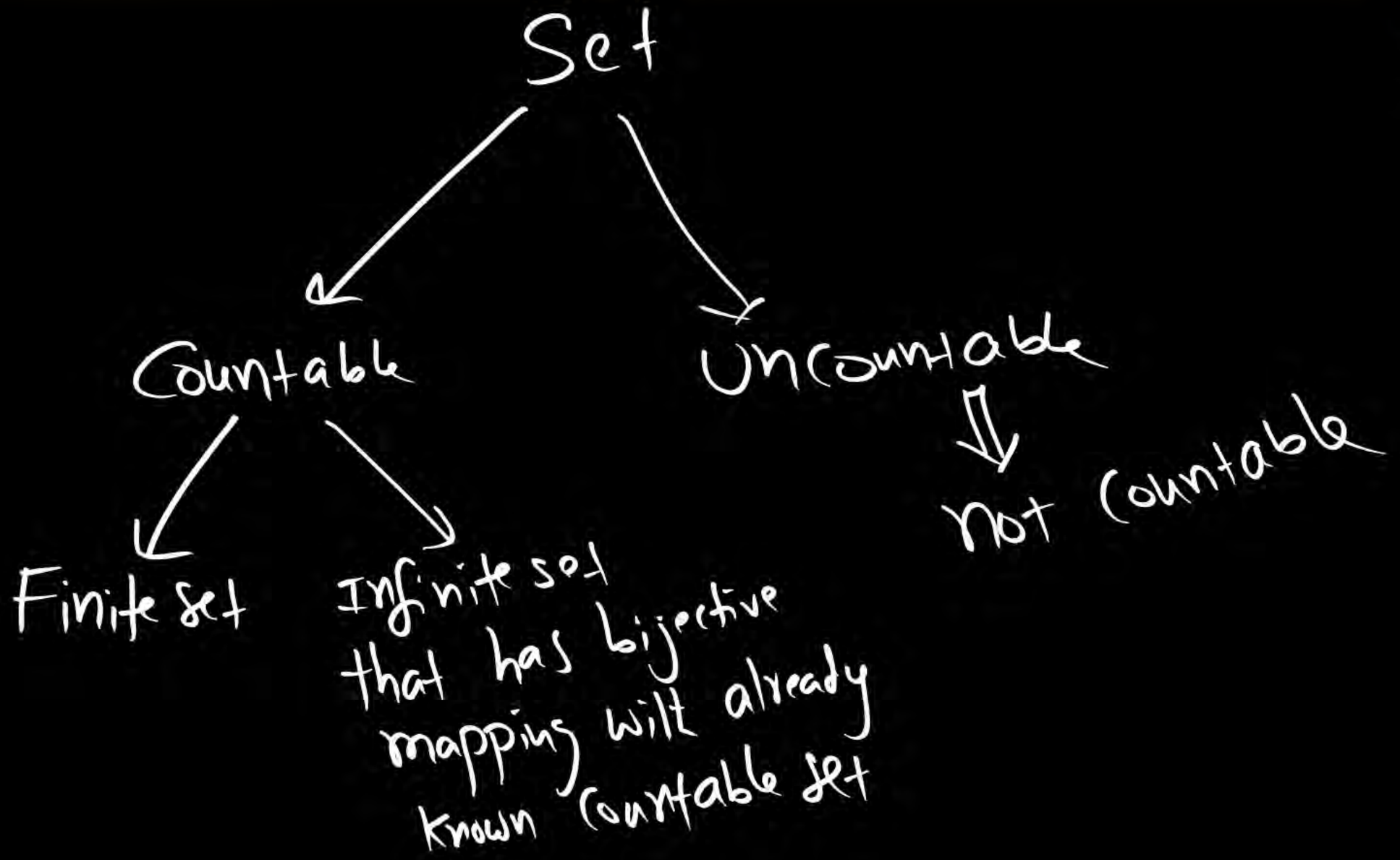
II) Every REL is countable set

III) Countable set need not be decidable

IV) Countable set need not be REL

V) Some countable sets are RELs

VI) " " " are not RELs


Set
Countable
Uncountable
Finite set
Infinite set that has bijective mapping with already known countable set
not countable

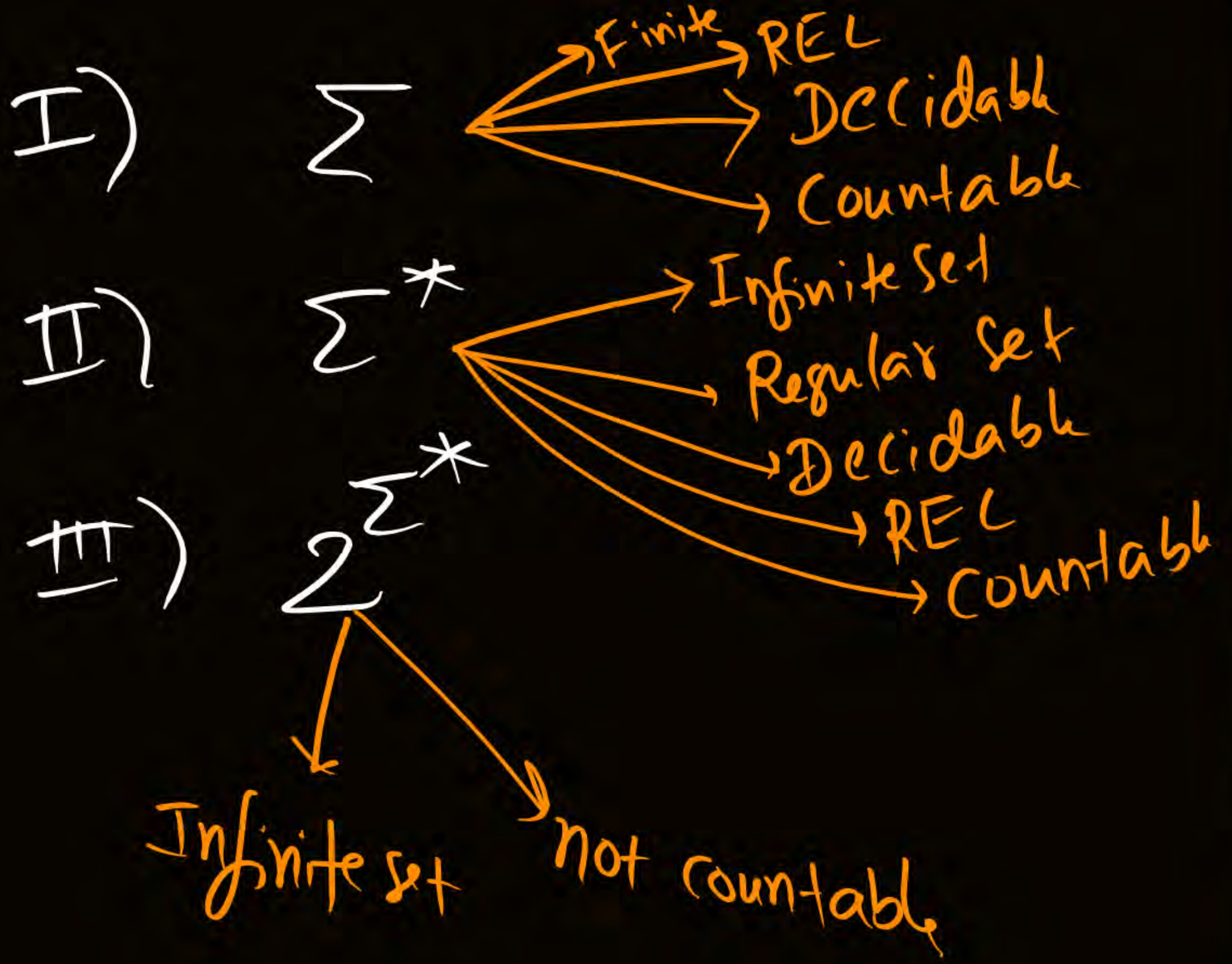
$\mathbb{N}$ = Natural set

Prime set

$\mathbb{Z}$ = Integer set

$2\mathbb{N}$ = Even set

$2\mathbb{N}+1$ = odd set



Σ = Set of finite no: of symbols

Σ^* = Set of all strings

2^{Σ^*} = Set of all languages

$$2^{\Sigma^*} = \{ L_1, L_2, L_3, L_4, \ldots \}$$

$$= \{ \{\}, \{\varepsilon\}, \{a\}, \{b\}, \{aa\}, \{ab\}, \ldots$$
$$\{\varepsilon, a\}, \{\varepsilon, b\}, \{\varepsilon, aa\}, \{a, b\}, \{a, aa\} \ldots$$
$$\{\varepsilon, a, b\}, \{\varepsilon, a, aa\}, \ldots$$
$$\vdots$$

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

	Regular Sets	DCFLs	CFLs	Recursive languages	RELs
① $L_1 \cup L_2$	Regular	CFL (need not be DCFL)	CFL	Recursive	REL
② $L_1 \cap L_2$	Regular	CSL	CSL	Recursive Set	REL
③ $\bar{L}$	Regular	DCFL	CSL	Recursive	either Recursive OR Not REL
④ $L_1 - L_2$ $= L_1 \cap \bar{L_2}$	Regular	CSL	CSL	Recursive	Any

Regular Languages $\subsetneq$ Infinite ...

DCFLs ✓ $CPIF_S$

CFLs ✗ $CIDSQI_{All}$

Recursive ✗ $SSHI_{All}$

REL ✗ QSI_{All}

Subset not closed for all

Infinite ...

h^{-1}

Finite subsets

closed for all

Turing M/C

$\cong$

Universal TM

$\cong$

Counter

$\cong$

2 Stack PDA

$\cong$ 2 way inf tape TM

$\cong$ multi tape TM

$\cong$ multi head TM

$\cong$ multi tape & multi head TM

$\equiv$ FA + ($\geq$ 2 stacks)

$\equiv$ PDA + ($\geq$ 1 stack)

Rice Theorm

↳ Every non trivial property is undecidable

non-trivial property

$L = \{ M \mid M \text{ is TM}, M \text{ accepts Regular} \}$ → Not RE

$\bar{L} = \{ M \mid M \text{ is TM}, M \text{ accepts not regular} \}$

Trivial

$L = \{TM \mid TM \text{ accepts REL}\}$

$= \text{Set of all TMs}$

$\bar{L} = \{TM \mid TM \text{ accepts not REL}\}$

$= \phi$

Church Turing Thesis :

↳ TM ≅ Computer

$L_1 \cap L_2$

Reg $\cap$ Reg $\Rightarrow$ Reg

CFL $\cap$ Reg $\Rightarrow$ CFL

CSL $\cap$ Reg $\Rightarrow$ CSL

Rec $\cap$ Reg $\Rightarrow$ Rec

REL $\cap$ Reg $\Rightarrow$ REL

$CFL \cap CFL \Rightarrow CSL$

$CFL \cap CSL \Rightarrow CSL$

$CFL \cap Rec \Rightarrow Rec$

$CFL \cap REL \Rightarrow REL$

$Rec \cap Rec \Rightarrow Rec$

$Rec \cap REL \Rightarrow REL$

REL ∩ REL ⇒ REL

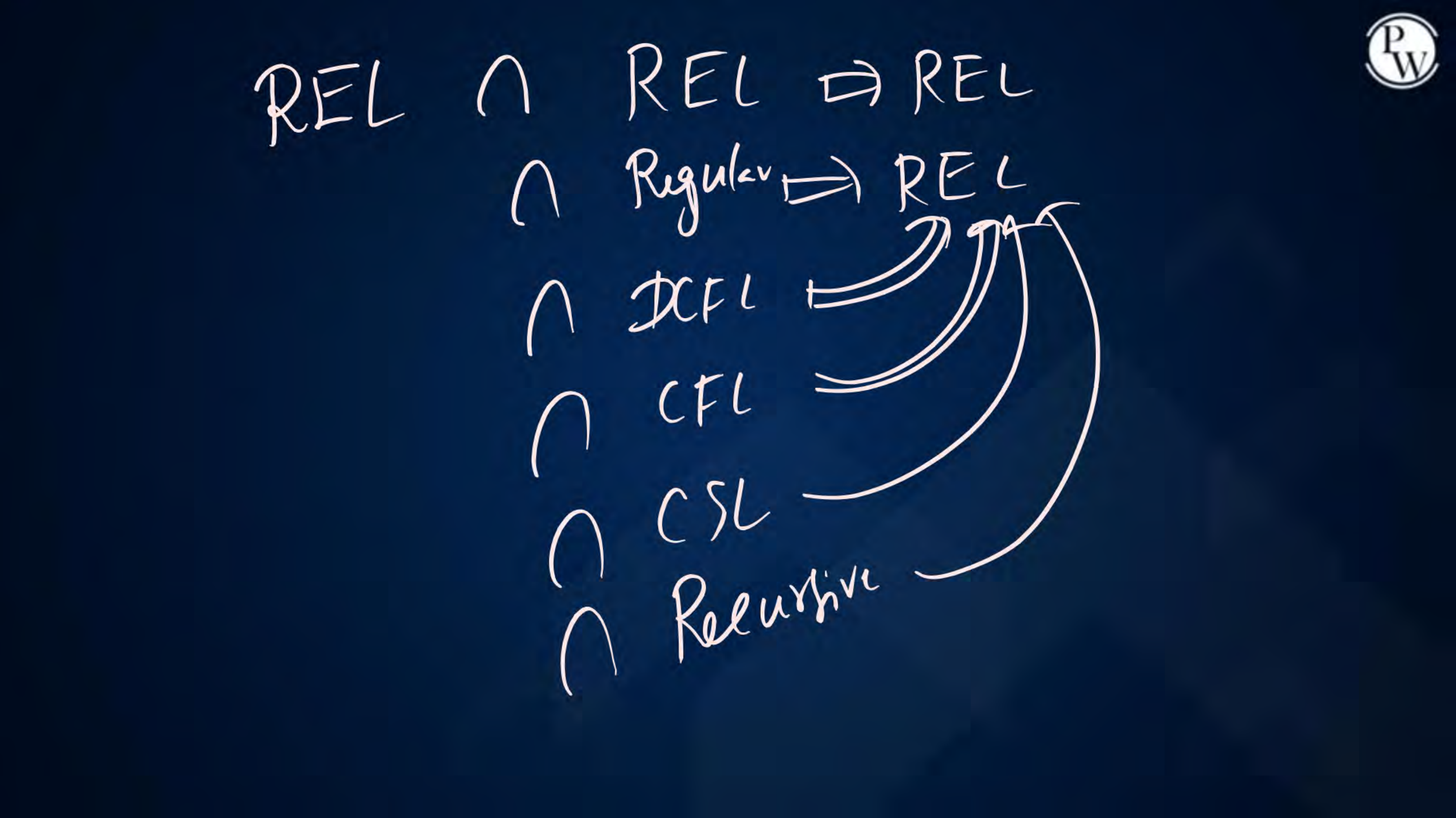
∩ Regular ⇒ REL

∩ DCFL

∩ CFL

∩ CSL

∩ Recursive



(Need not be Regular)

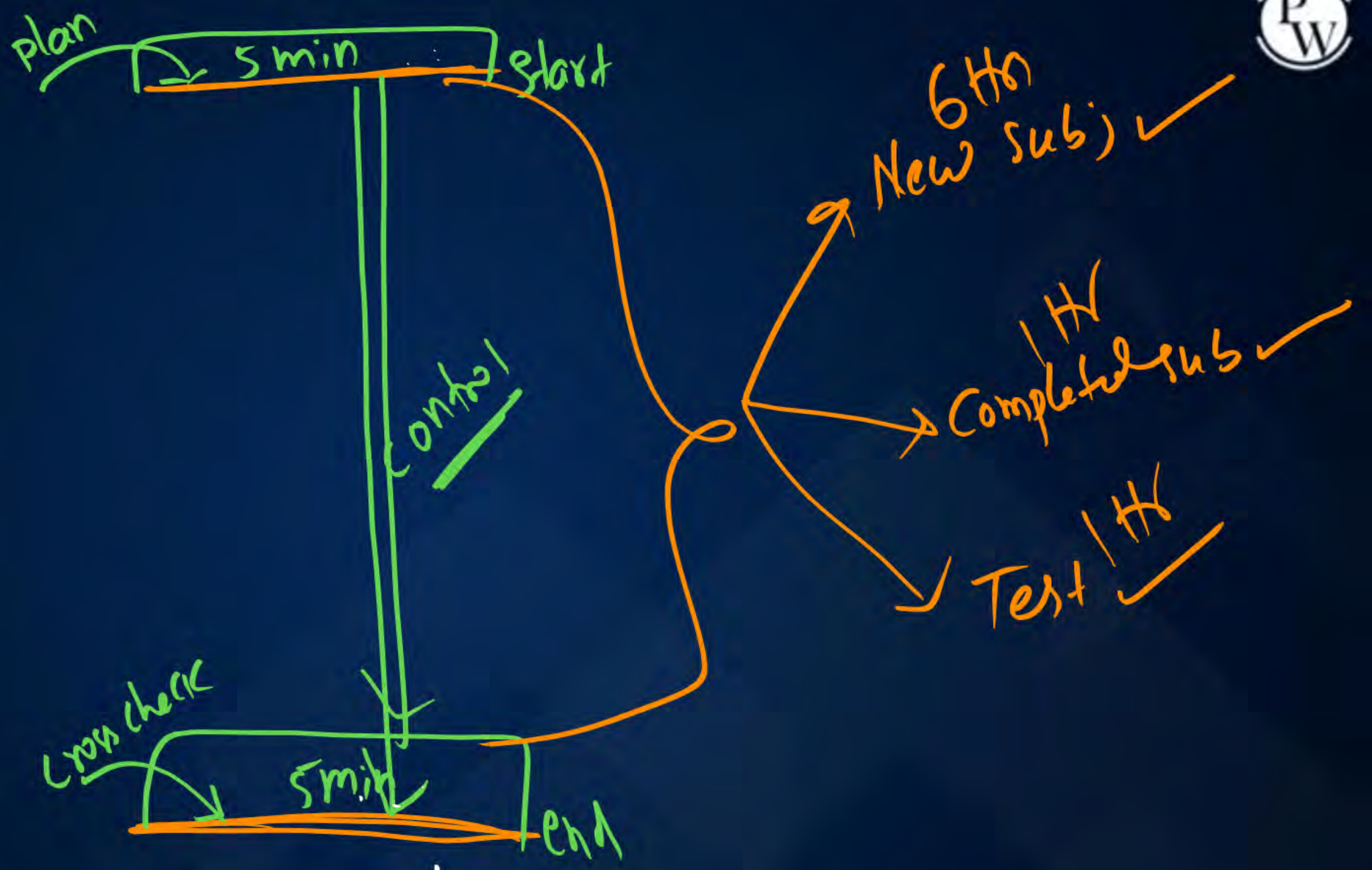
$$Reg \cap DCFL \Longrightarrow DCFL$$

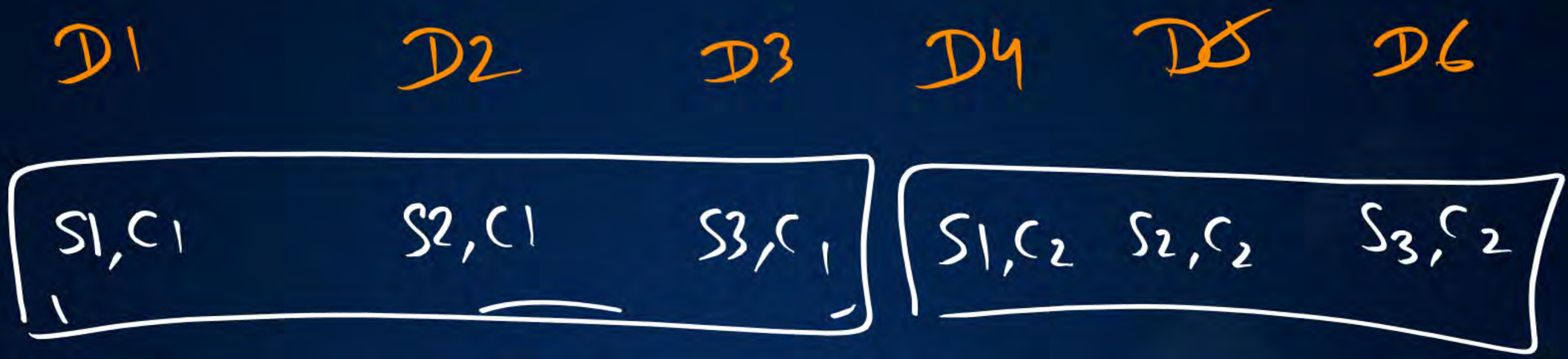
Case 1: $(a+b)^* \cap (a+b)^* \longrightarrow (a+b)^*$

Reg ∩ DCFL

Case 2: $(a+b)^* \cap a^n b^n \longrightarrow a^n b^n$

nonreg DCFL

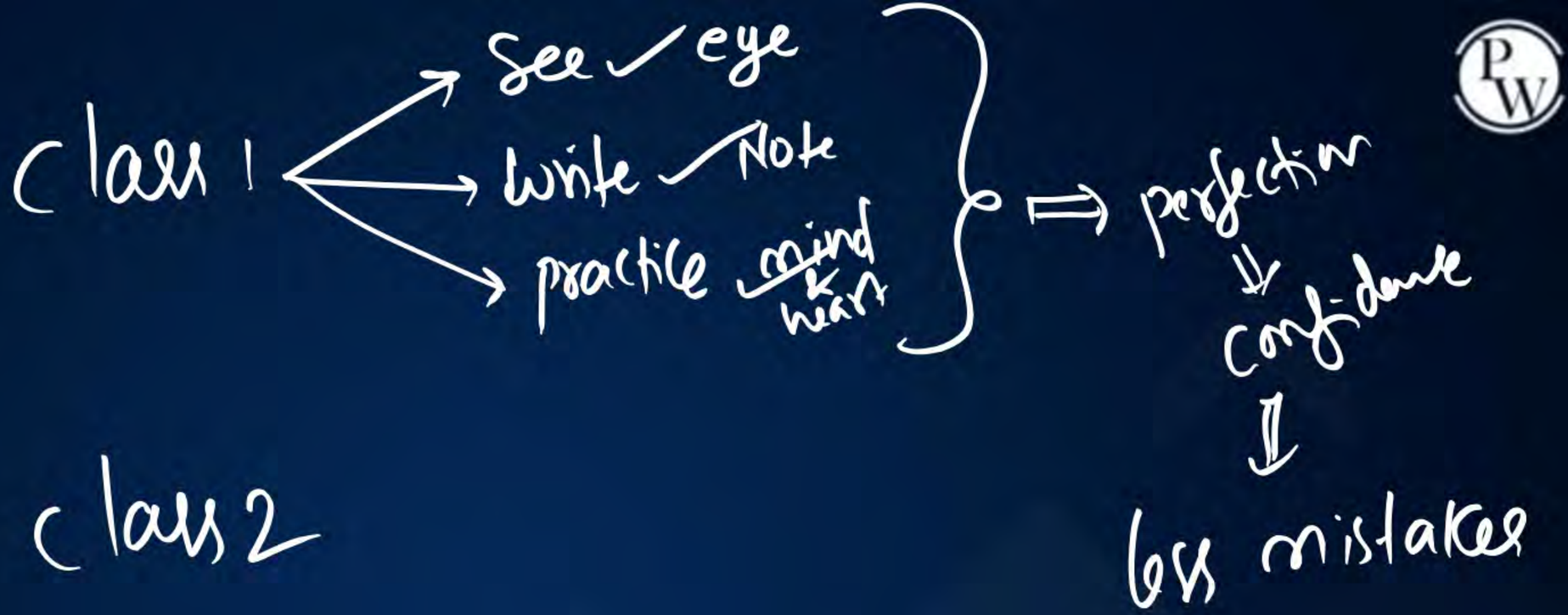

plan
5 min
start
6th
New Subj
1 Hr
Completed Subj
Test 1 Hr
Control
Cross check
5 min
End


D1
D2
D3
D4
D5
D6
S1, C1
S2, C1
S3, C1
S1, C2
S2, C2
S3, C2

Backlog ?

~~Completion~~

Perfection & Target Systems } ⇒ ✓


class 1
See ✓ eye
write ✓ Note
practice mind & heart
⟹ perfection
confidence
less mistakes
class 2


Mistakes
marks
1st :
70 %.
low
2nd :
50 %.
med
3rd :
10 %.
Better
60 — 80 %
Marks

Summary

Q1

1st: Faculty ⇒ Q1 ⟹ (watch) understanding

2nd: Student ⇒ Q1 ⟹ (practice) ?


THANK YOU
GW
SOLDIERS !