1) Halting Problem:

D

FA:

DPDA

PDA

LBA

HTM

For given FA and given string w :

"whether FA halts on w."

Halting problem for FA/DPDA/PDA/LBA/HTM is Decidable

UD

TM:

Logic exist

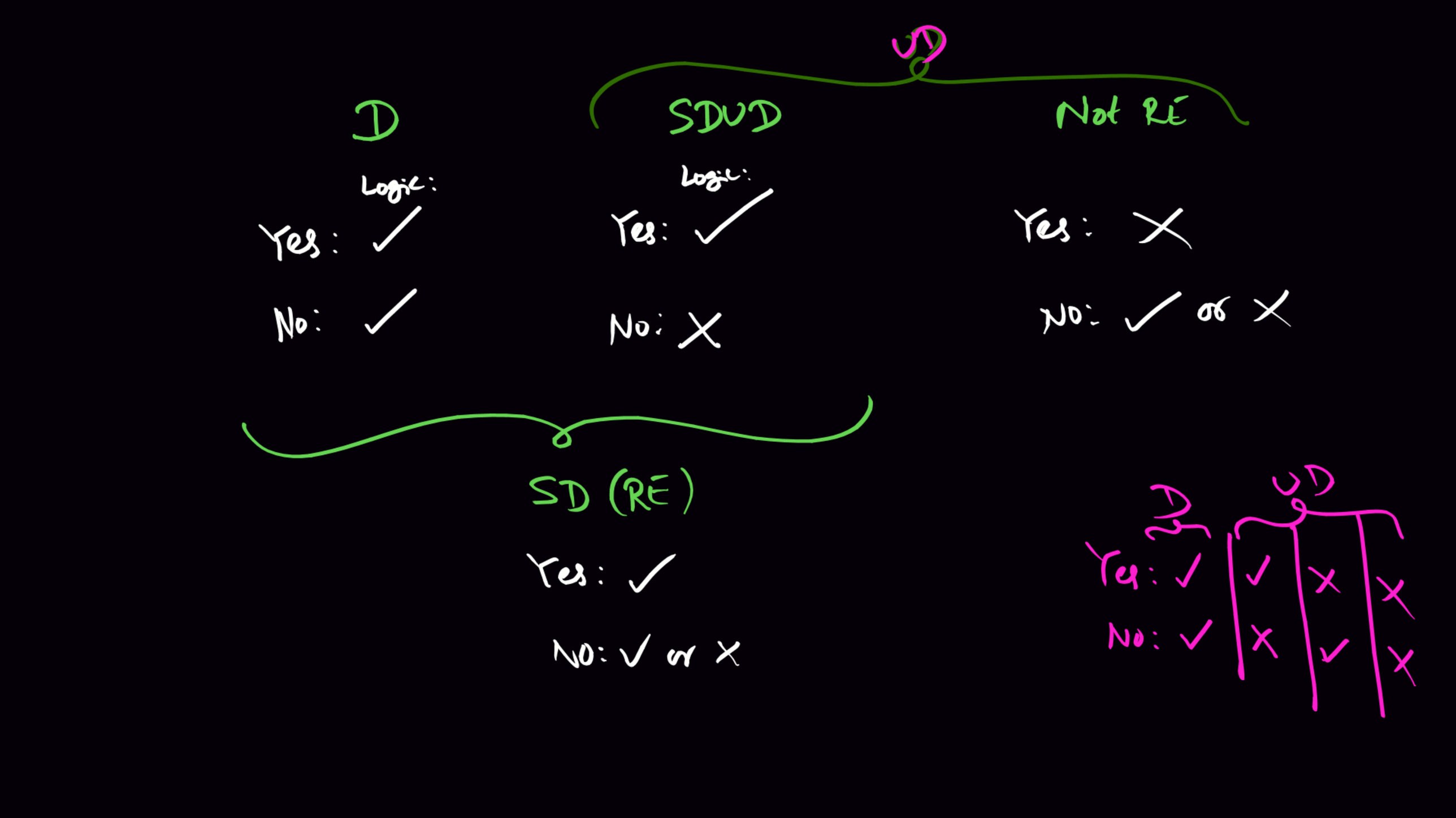
TM halts on w.

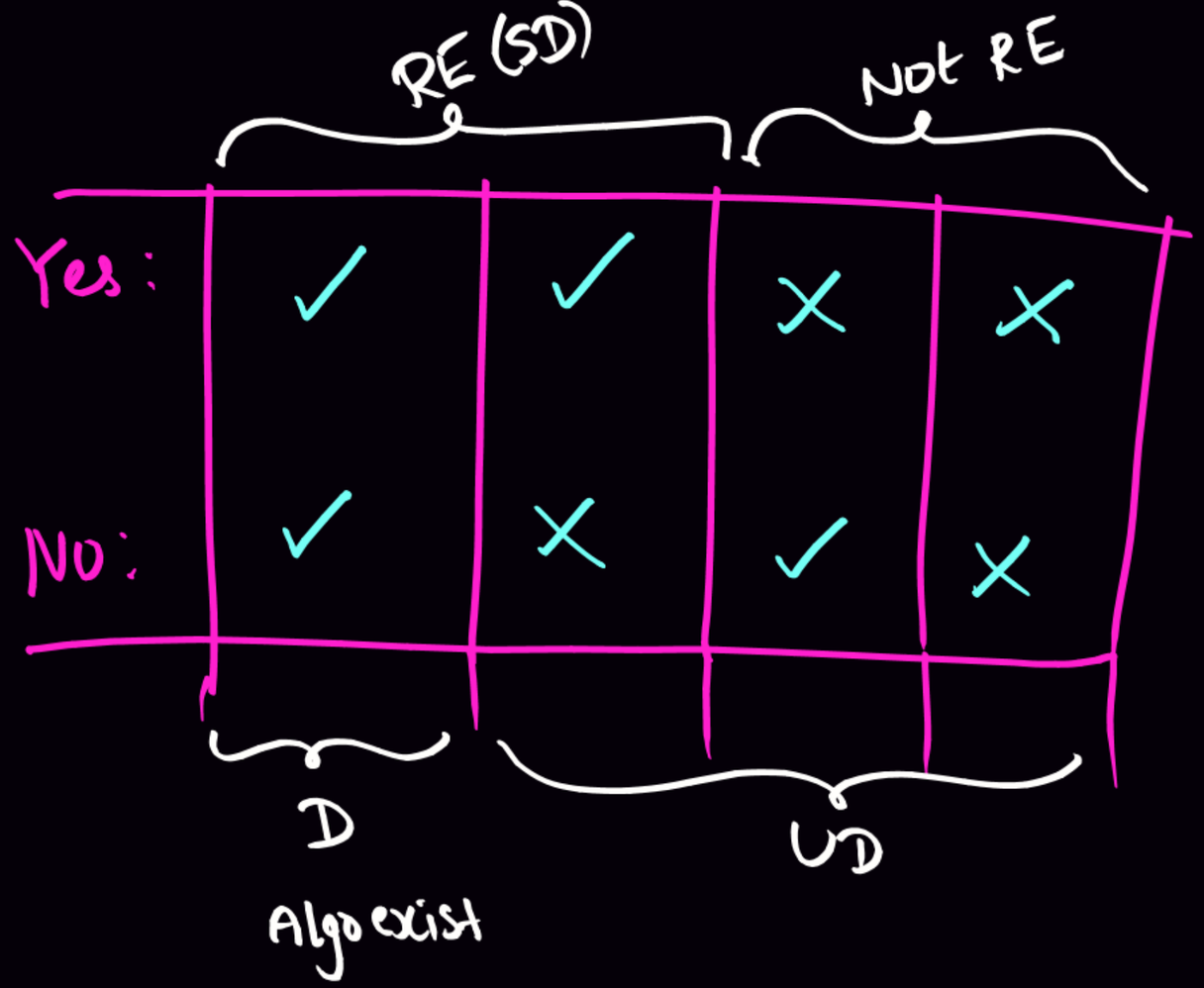
TM doesn't halt on w.

Logic not exist

RE but not rec

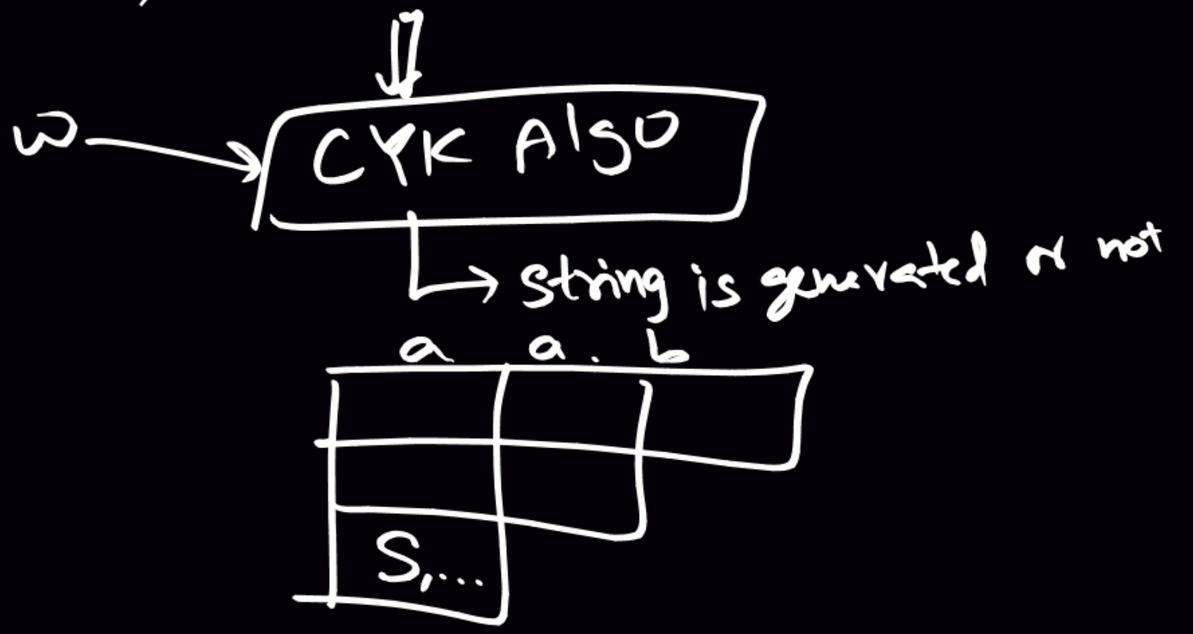
Halting problem for TM is SDUD.


D
Logic:
Yes: ✓
No: ✓
SDUD
Logic:
Yes: ✓
No: ✗
UD
Not RE
Yes: ✗
No: ✓ or ✗
SD (RE)
Yes: ✓
No: ✓ or ✗
D
SD
UD
Yes: ✓
No: ✓
✓ ✗ ✗
✗ ✓ ✗


RE (SD)
Not RE
Yes:
No:
D
UD
Algo exist

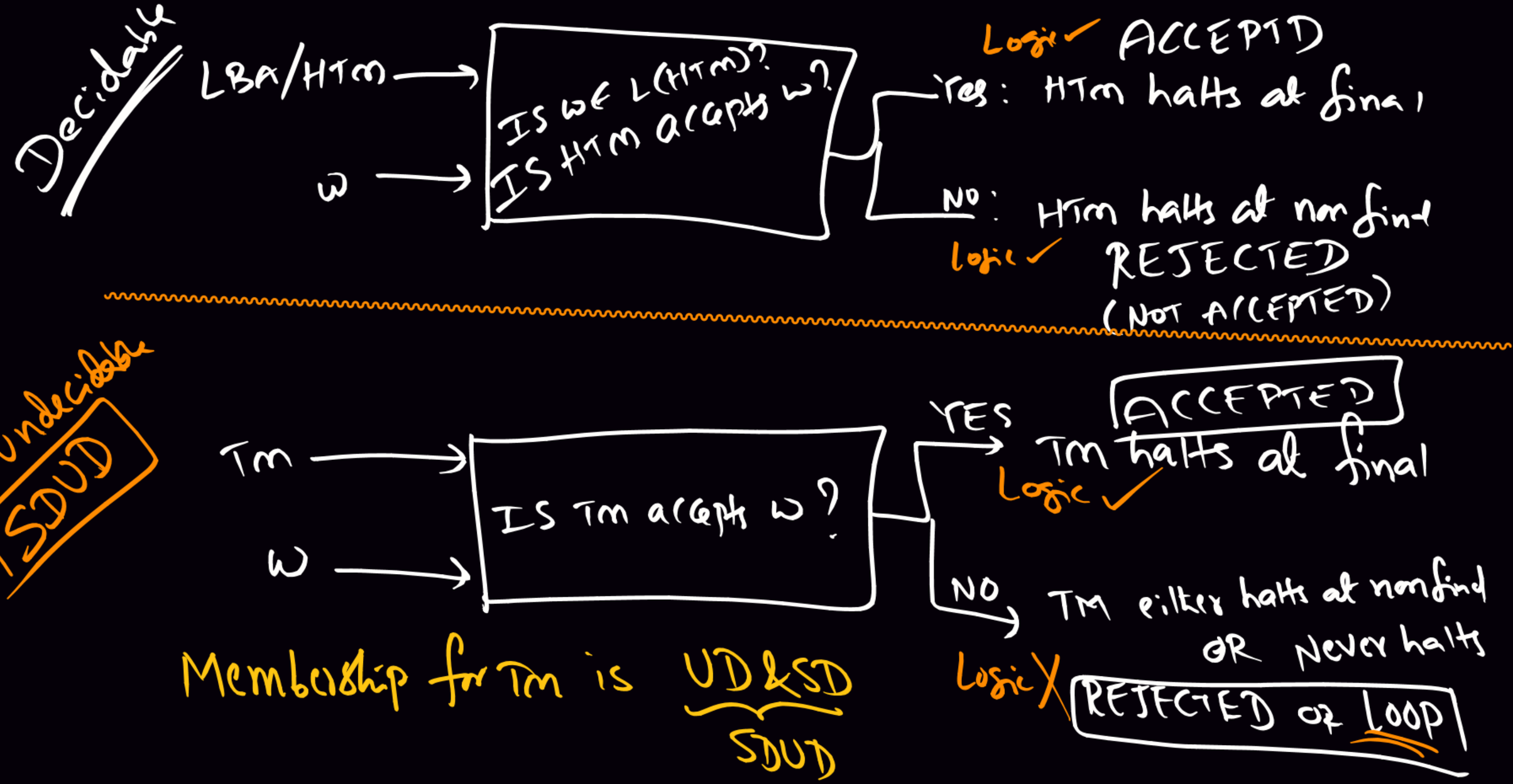
2) Membership:

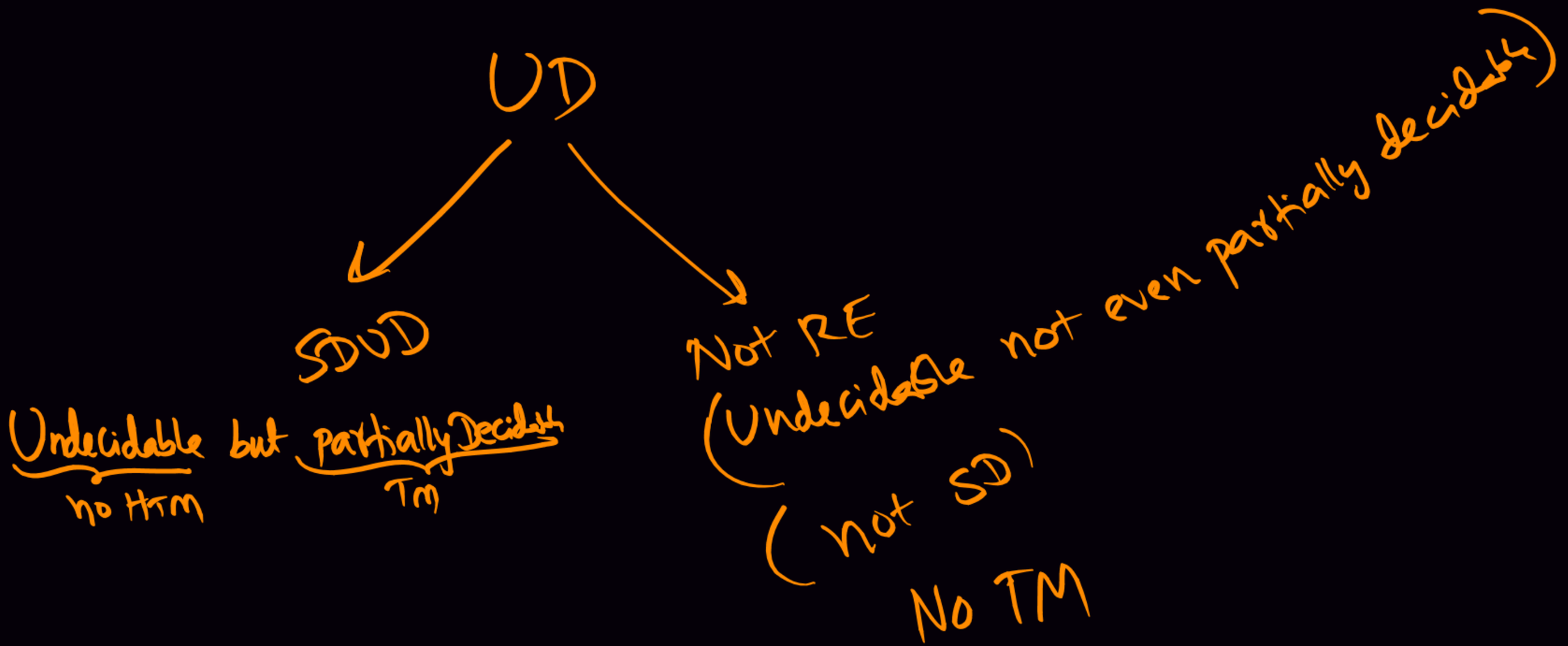
Decidable: FA/DPDA/PDA $\Longrightarrow$ CFG $\Longrightarrow$ CNF CFG



CYK Algorithm

Bottom up parsing
Membership algo
$O(n^3)$
Dynamic Programming


Decidable
LBA/HTM
w
Is w ∈ L(HTM)?
Is HTM accepts w?
Logic ✓ ACCEPTD
Yes: HTM halts at final
NO: HTM halts at non final
Logic ✓ REJECTED
(NOT ACCEPTED)
Undecidable
SDUD
TM
w
IS TM accepts w?
YES
ACCEPTED
TM halts at final
Logic ✓
NO
TM either halts at non final
OR Never halts
Logic X
REJECTED or LOOP
Membership for TM is UD & SD
SDUD


UD
SDUD
Undecidable but partially Decidable
no HTM
TM
Not RE
(Undecidable not even partially decidable)
(not SD)
No TM

1) Is FA accepts ab ? ⇒ Membership for FA : Decidable

2) Is FA halts on ε ? ⇒ Halting for FA : "

3) Is FA_1 equivalent to FA_2 ? ⇒ Equivalence for FA : "

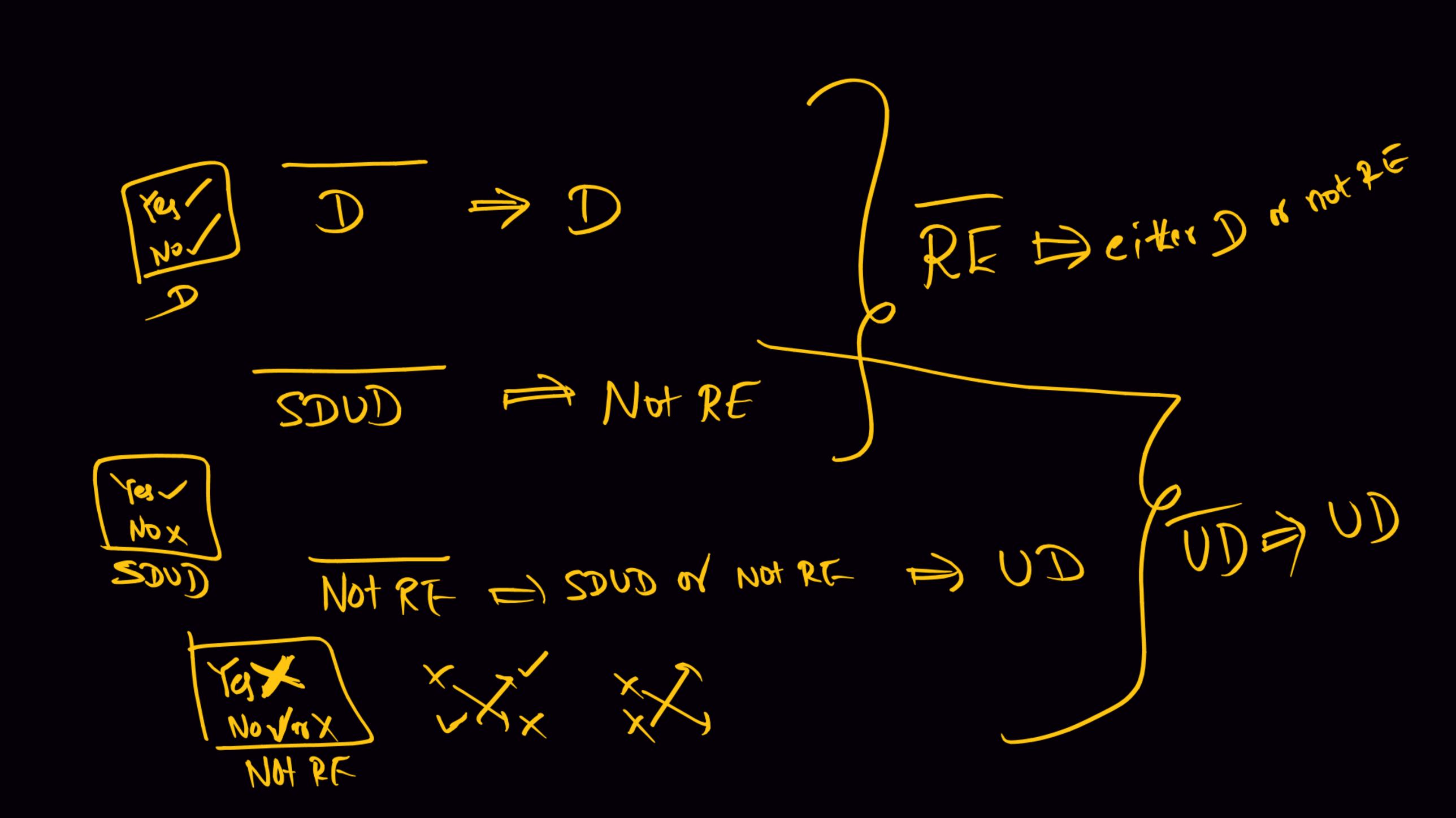
4) Is FA equivalent to Reg Exp ?

5) Is FA accepts finite lang ? ⇒ Finiteness for FA : decidable

6) Is FA accepts nothing ? ⇒ Emptiness for FA : "

7) Is FA accepts everything ? ⇒ Totality for FA : "

8) Is FA accepts something ? ⇒ Non-emptiness for FA : "


Yes ✓
No ✓
D
$\overline{D} \Rightarrow D$
$\overline{RE} \Rightarrow$ either D or not RE
$\overline{SDUD} \Rightarrow$ Not RE
Yes ✓
No ✗
SDUD
$\overline{\text{Not RE}} \Rightarrow$ SDUD or Not RE $\Rightarrow$ UD
$\overline{UD} \Rightarrow UD$
Yes ✗
No ✓ or ✗
Not RE

Undecidable

9) IS Tm accepts ε ? ⇒ Membership for Tm ⇒ SDUD - (RE but not Rec)

10) IS Tm halts on ab ? ⇒ Halting for Tm ⇒ SDUD

11) IS Tm accepts a^* ? ⇒ Not RE

12) IS Tm accepts {ε, a, ab} ?
- YES : Tm should accept only ε, a, and ab
  Logic not exist
  $L(Tm) = \{\varepsilon, a, ab\}$
- NO :

⇒ Not RE

13) IS TM accepts only ab ? IS $L(Tm) = \{ab\}$? ⇒ Not RE

14) IS Tm accepts nothing ? IS $L(Tm) = \phi = \{\}$? ⇒ Not RE

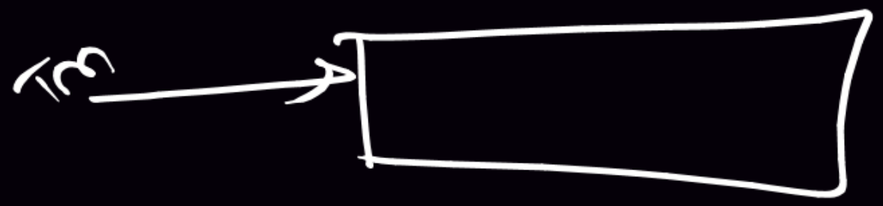
15) IS Tm accepts something ? IS TM accepts some string ? ⇒ SDUD

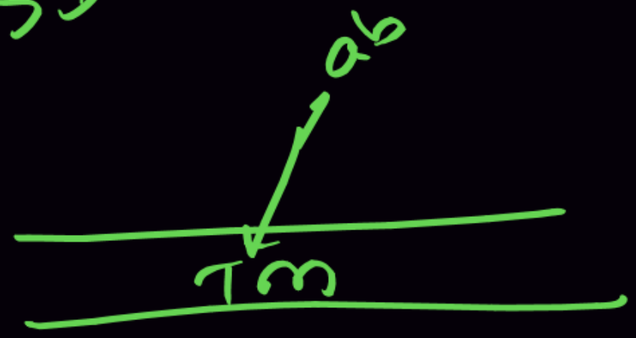
16) IS Tm accepts everything ! ⇒ Not RE

IS $L(Tm) = \Sigma^*$?

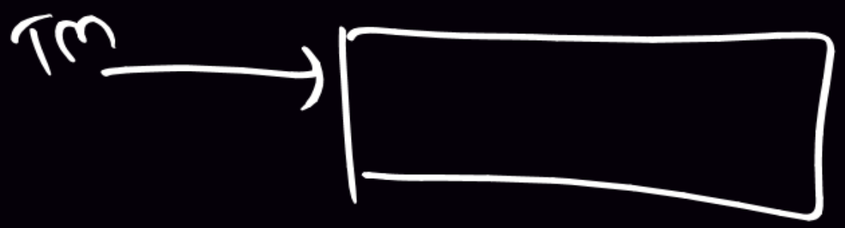
ε : Tm should accept
a : "
ab : "
Remaining all should not accept

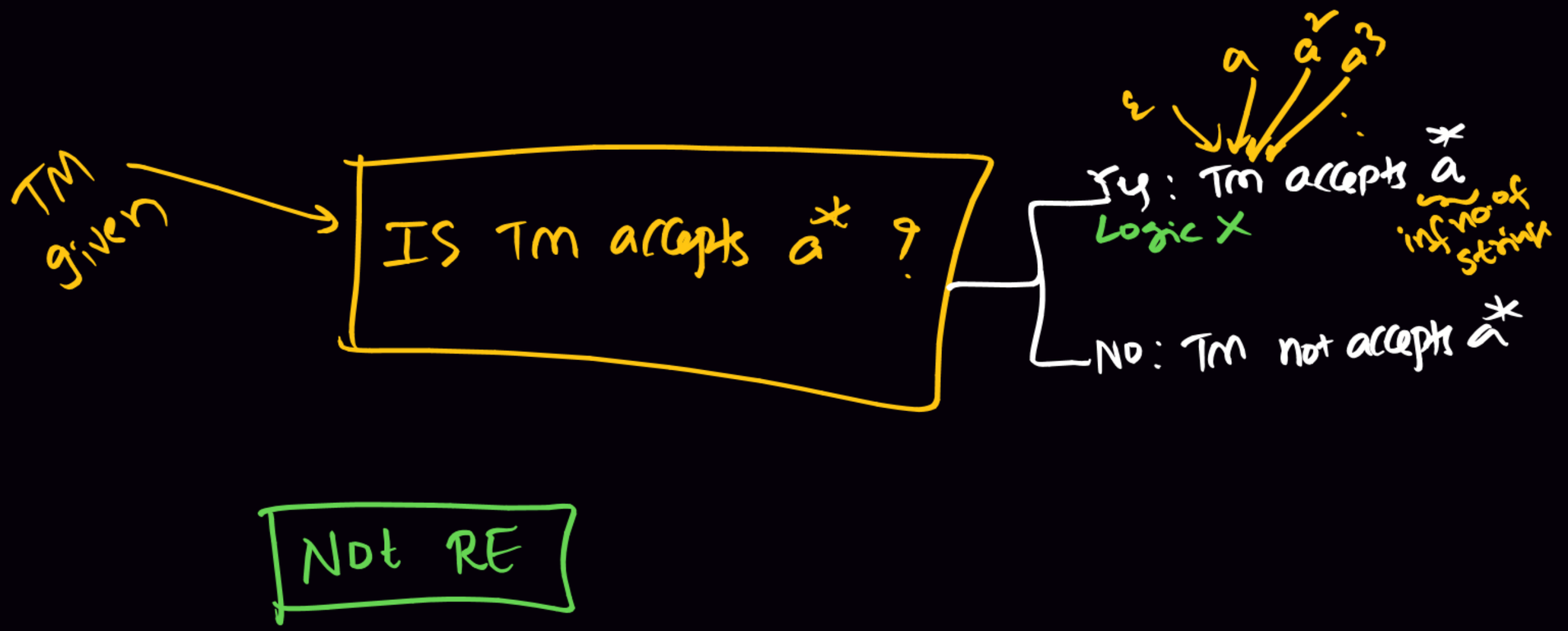
IS given TM accepts ab ? ⇒ SDUD





IS given TM accepts Something ? ⇒ SDUD




TM given
IS TM accepts a^* ?
ε
a
a^2
a^3
Yes: TM accepts a^*
Logic X
inf no of strings
No: TM not accepts a^*

Not RE

17) Is Tm accepts REL ? Always YES (Trivial) ⇒ Decidable

18) Is Tm accepts not REL ? ⇒ Always NO (Trivial) ⇒ Decidable

*** 19) Is Tm accepts Regular ? ⇒ Not RE ⇒ UD

~ (Non-trivial)

20) Is Tm accepts CFL ?

21) Is Tm accepts $a^n b^n$?

22) Is Tm accepts Recursive ?

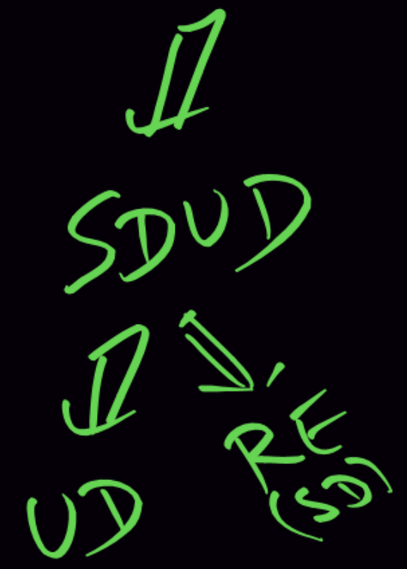
23) Is Tm accepts CSL ?

$\Rightarrow$ Not RE

$\Rightarrow$ UD

24) Whether TM reaches State q on given String ab.

(TM reaches q on w)

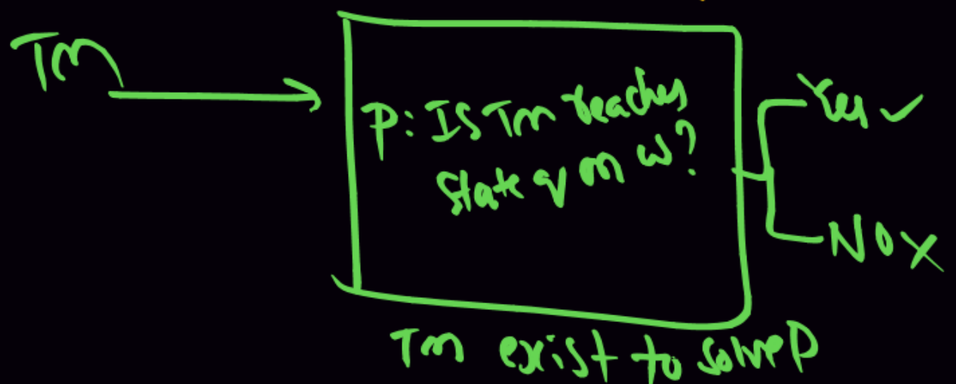


→ Yes : TM will reach state q on w

Logic ✓

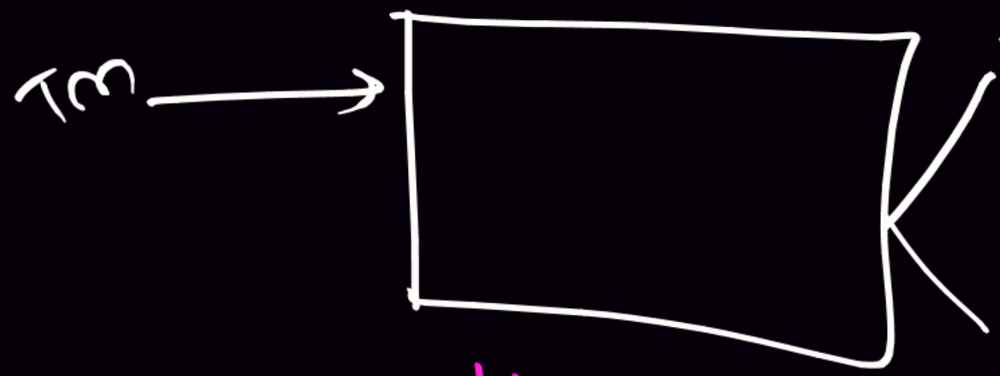
→ No : TM will not reach state q on w

Logic ✗



***25) IS TM reaches state q on w within 100 steps?

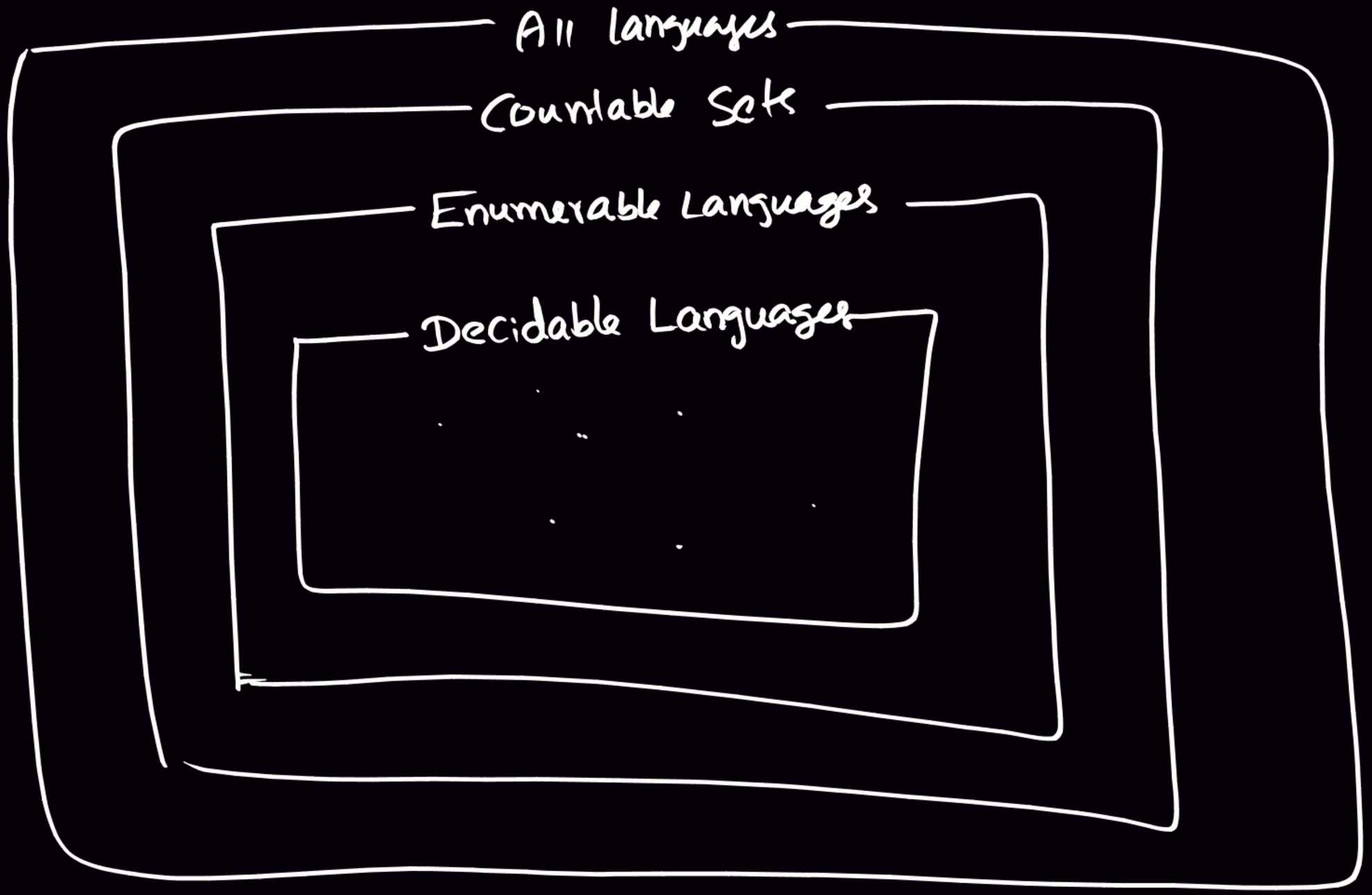
↳ Decidable



Yes: TM reaches q on w within 100 steps.
Logic ✓

Logic ✓
NO: TM did not reach q on w within 100 steps.

1 —(step / move / Transition)→ 2 → 3 → 1 → 2 → 2 → 4 → . . . → 100th step → Stop


All languages
Countable Sets
Enumerable Languages
Decidable Languages

1) Every Decidable set is Enumerable

2) " " " is Countable

3) Every Enumerable set is Countable.

4) Enumerable lang is need not be decidable.

5) Countable lang is need not be enumerable

6) " " " decidable.

Reducibility:

$A \leq B$

$A \propto B$

$A \leq_m B$

$A \propto_p B$

A is reducible to B

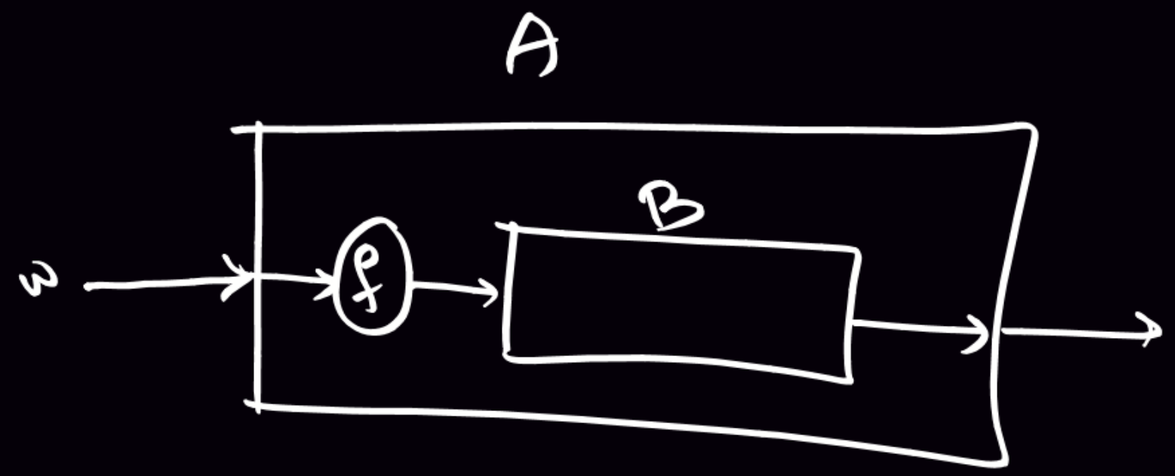
A is many to one reducible to B

A is polynomially reducible to B

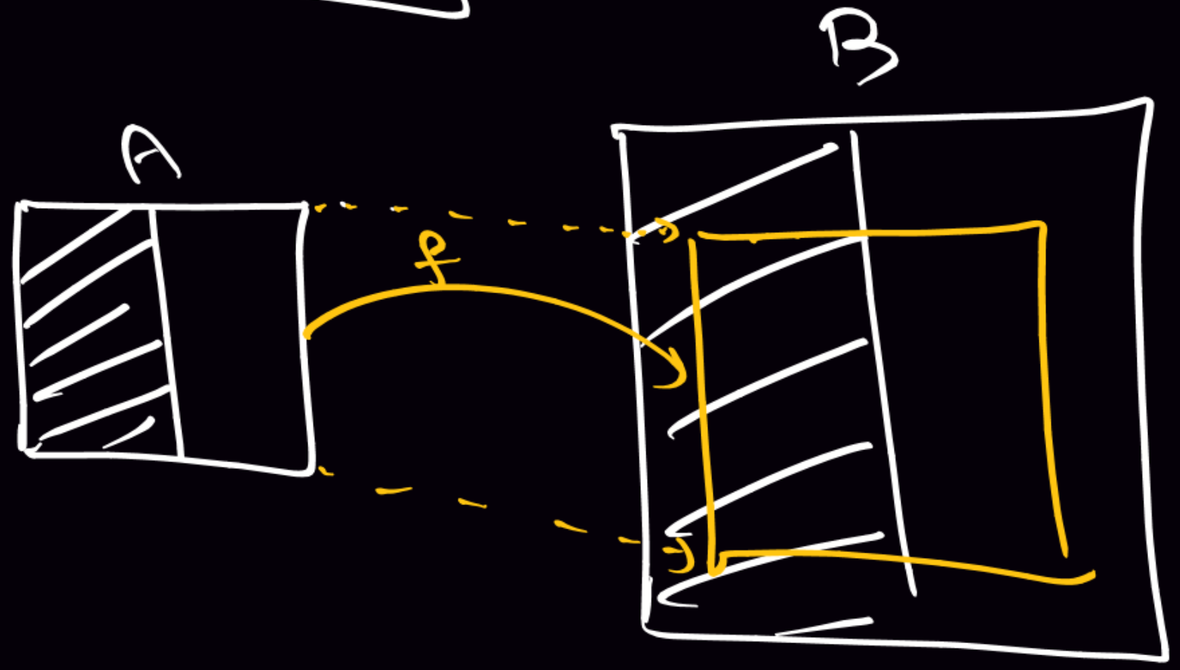
B is atleast as hard as A

$\text{Swap} \overset{A}{\leq} \overset{B}{\text{Sort}}$

$\text{Sort}_1 \leq \text{Sort}_2$



To solve A,
convert instances of A
into instances of B.

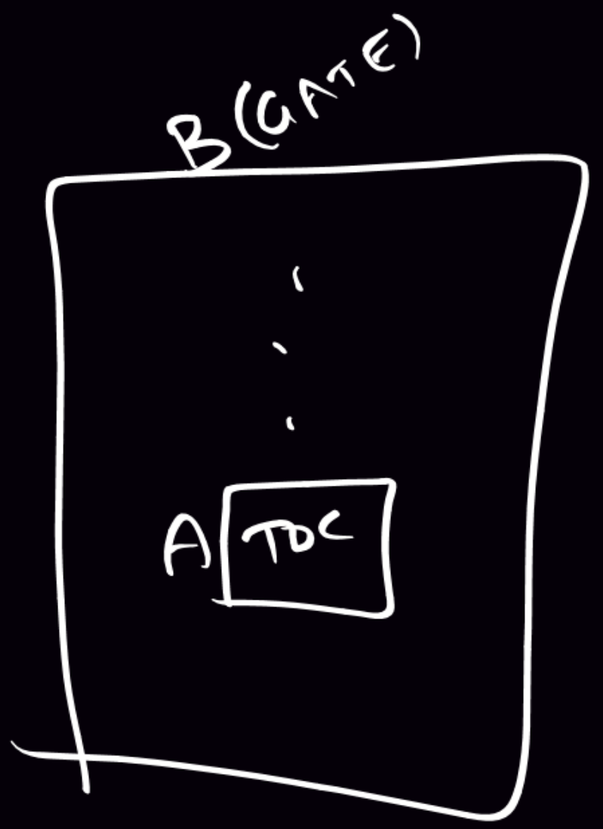


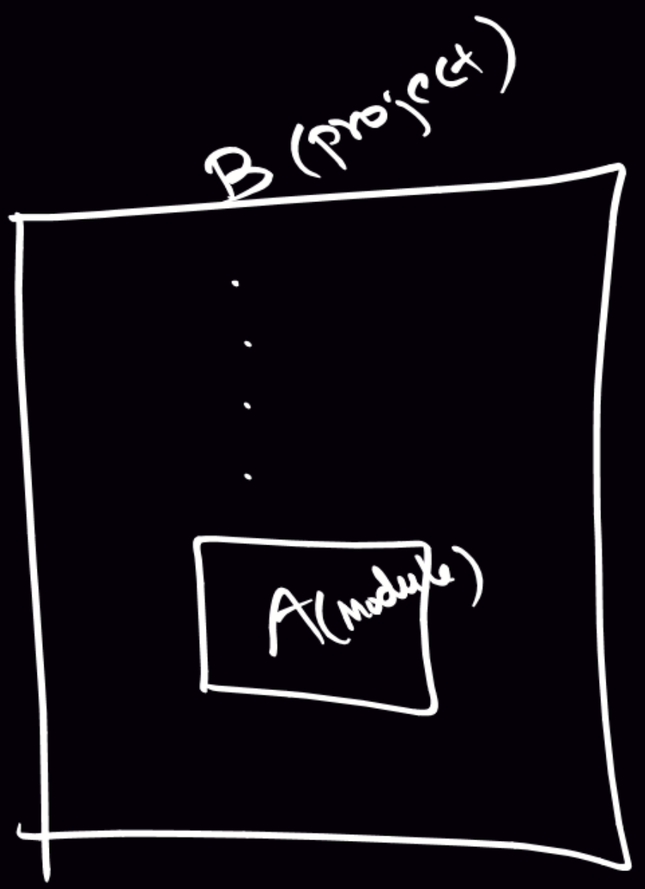
$A \leq B$

I) A and B are equal harder

OR

II) B is more harder than A





Let $A \leq B$:

1) If B is Decidable then A is Decidable
2) If A is Undecidable then B is Undecidable
3) If A is Decidable then B is either D or UD
4) If B is Undecidable then A is either D or UD
5) If A takes 100 days then B takes Min 100 days
6) If B takes 100 days then A takes Max 100 days

7) If B is RE then A is RE

8) If A is not RE then B is not RE

If $A \leq B$ and $B \leq A$

then

i) If A is Decidable then B is Decidable

ii) If B is Decidable then A is Decidable