CS & IT ENGINEERING

Theory of Computation

Undecidability &
Decidability :Practice session

Lecture 03

By- DEVA Sir

01 Decidable Languages

02 RE but not recursive Languages

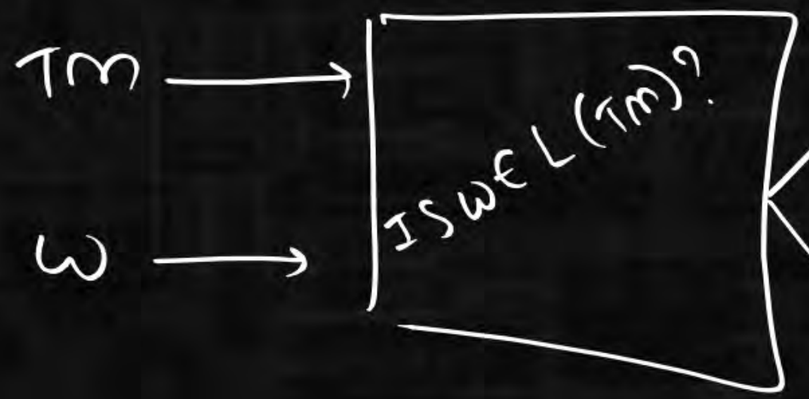
03 Not RE Languages


FA
DPDA
PDA
HTM
TM
H
M
E
F
T
E
D
S
Yes: logic
No: logic
Yes:
No:
Yes:
No:
Yes:
No:
Yes:
No:

Membership for TM

- RE but not RC
- Undecidable
- RE

↳ Is given TM accepts given string ?



Yes : TM accepts w
$w \in L(TM)$

Logic exist

given TM should halt at final for given string

No : TM doesn't accept w
$w \notin L(TM)$

Given TM either halt at non final or never halts for given string

logic not exist

Check L_i is Decidable (D), RE but not ReC (SDUD), or not RE (NR)

1. $L_1 = \{a^n b^n\}$
2. Every Finite Language over Σ
3. Every Regular language over Σ
4. Every DCFL
5. Every CFL
6. Every CFL but not DCFL
7. Every CSL
8. $a^* b^*$
9. $a^n b^n c^n$
10. $\{www \mid w \in \{a,b\}^*\}$
11. $\{a^{\text{Prime}}\}$
12. $\{a^{n^n}\}$

$\Rightarrow$ D

D, SDUD, NR

(13) $\{F \mid F \text{ is finite language over } \Sigma\}$

(14) $\{R \mid R \text{ is Regular set}\}$

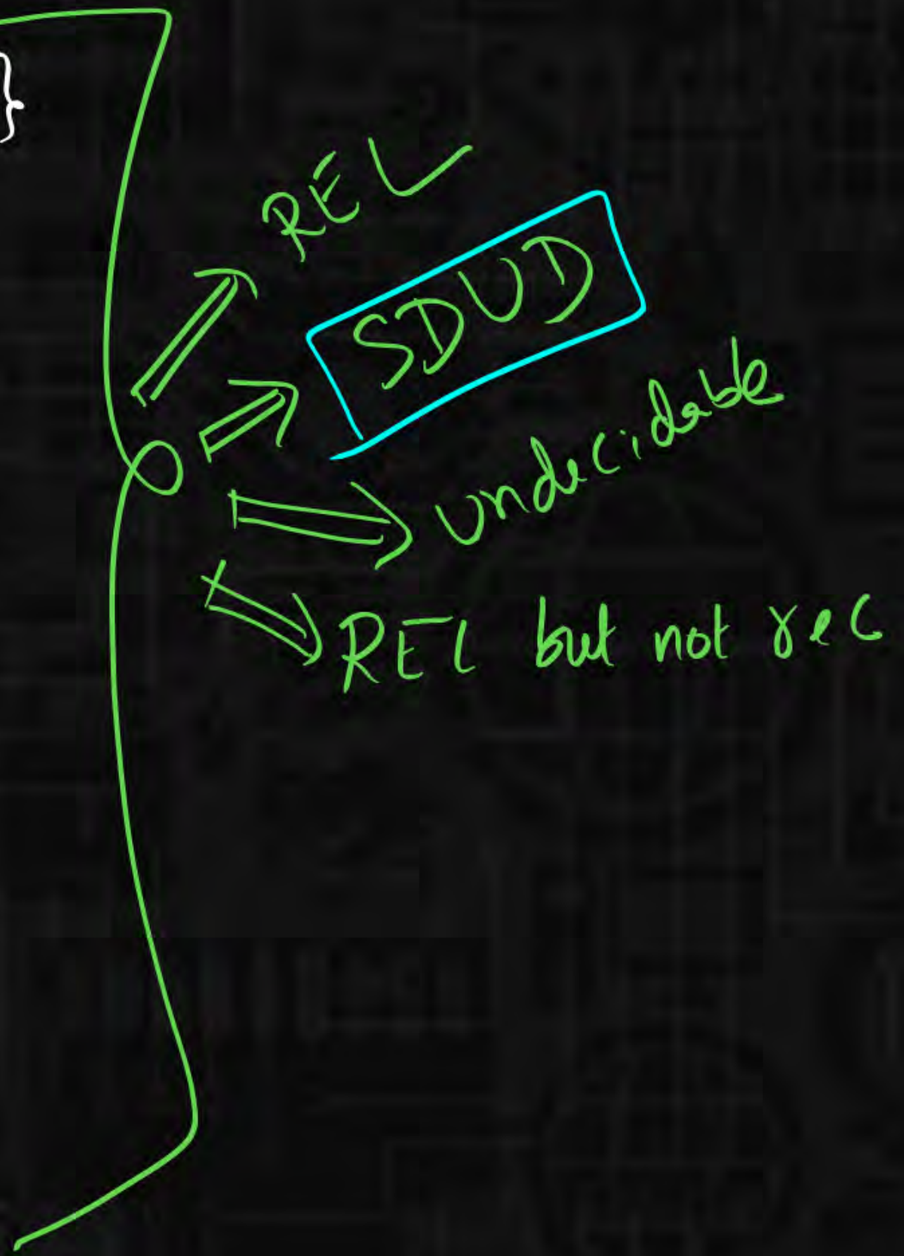
(15) $\{L \mid L \text{ is CFL}\}$

(16) $\{L \mid L \text{ is CSL}\}$

(17) $\{L \mid L \text{ is recursive}\}$

(18) $\{L \mid L \text{ is REL}\}$

(19) $\{L \mid L \text{ is RE but not rec}\}$



(13) $\{F \mid F \text{ is finite lang}\}$ = Set of all finite languages ⇒ UD

L_{13} = Set of all finite languages

TM exist

$= \{F_1, F_2, F_3, F_4, F_5, F_6, \ldots\}$ ⇒ TM exist (HTM not exist)

F_1: TM, F_2: TM, F_3: TM, F_4: TM, F_5: TM exist, F_6: TM exist

Every valid member has logic

UD

$\overline{L_{13}}$ = Set of all infinite languages

TM not exist

$= \{I_1, I_2, I_3, I_4, I_5, I_6, \ldots\}$ ⇒ No TM

I_1: TM exist, I_2: TM exist, I_3: TM exist, I_4: not REC ⇒ logic not exist, I_5: TM exist, I_6: TM exist

(20) $\overline{L_{13}}$ = Set of all not finite languages

(21) $\overline{L_{14}}$ = Set of all not regulars

(22) $\overline{L_{15}}$ = Set of all not CFLs

(23) $\overline{L_{16}}$ = Set of all not CSLs

(24) $\overline{L_{17}}$ = Set of all not decidables

(25) $\overline{L_{18}}$ = Set of all not RELs

(26) $\overline{L_{19}}$ = Set of all languages which are recursive, or not RELs

$L = \{L_1, L_2, [L_3], \}$

not REL
(logic not clear)

$\}$ → not REL

(27) $\{TM \mid L(TM) = \{\}\}$ ⇒ Not RE

Emptiness

(28) $\{TM \mid L(Tm) = \{\varepsilon\}\}$ ⇒ Not RE

(29) $\{TM \mid L(Tm) = \{0, 11, \varepsilon\}\}$

(30) $\{TM \mid L(Tm) = (01)^*\}$

(31) $\{Tm \mid L(Tm) \neq \phi\}$ ⇒ RE but not (SD UD)

non emptiness

(32) $\{Tm \mid L(Tm) = \text{Infinite}\}$ ⇒ Not RE

TM accepts ε ?

↳ Yes: TM halts at final on ε

Logic exist

TM accept only ε ?

Logic not exist

↳ Yes: L(TM) = {ε}

✓ 1st: TM should accept ε

? 2nd: TM should not accept any other

Problem

↳ 1st find logic for YES

If logic not exist ⟹ Answer is over

Not REC

If logic exist for YES

↳ 2nd: find logic for NO

Non-trivial

(33) $\{ \langle TM \rangle \mid TM \text{ accepts finite lang} \}$ ⇨ Not REL

finiteness

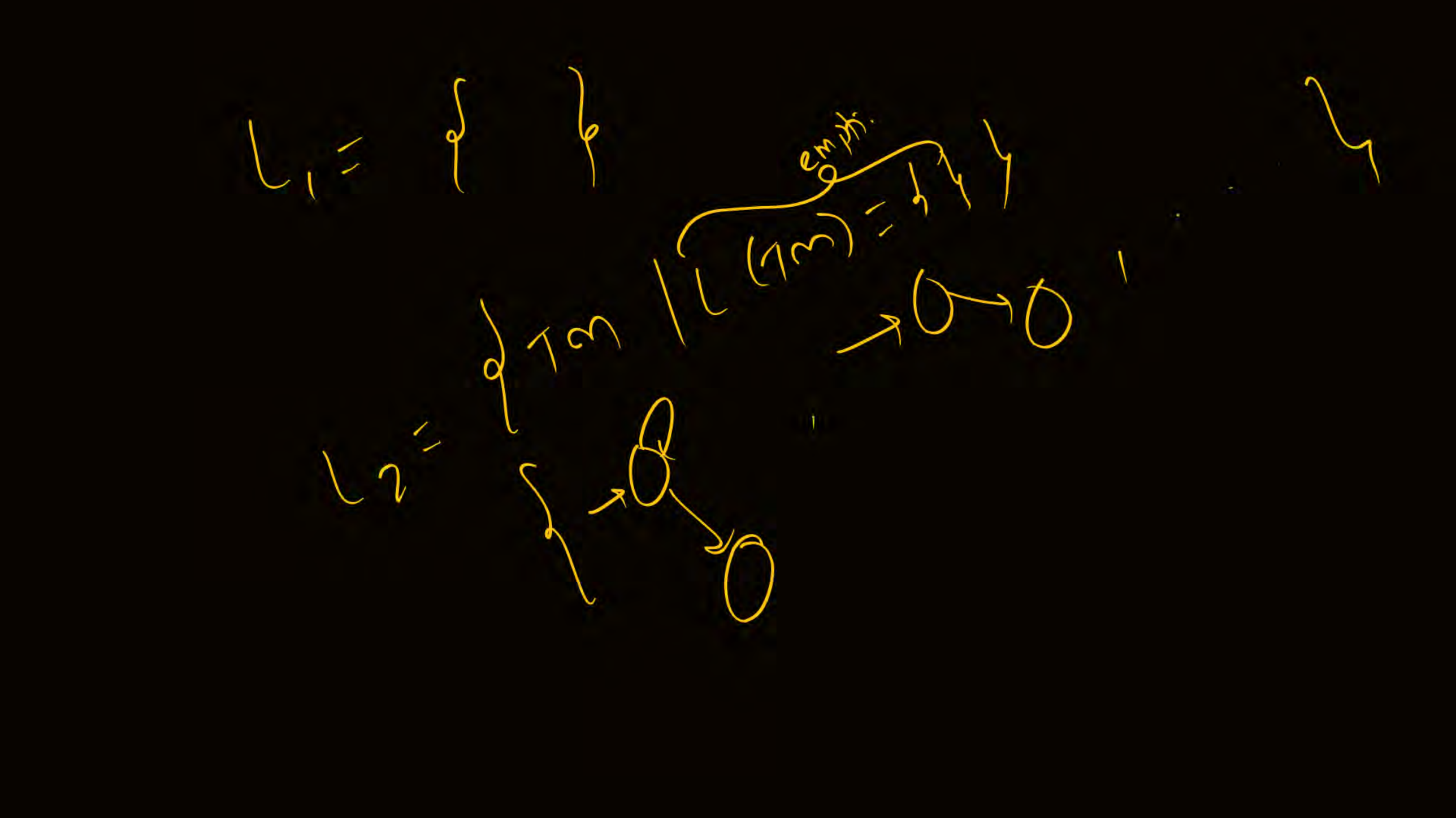
(34) $\{ \langle Tm \rangle \mid Tm \text{ accepts regular} \}$

(35) $\{ \langle Tm \rangle \mid Tm \text{ accepts recursive} \}$

***(36) Trivial $\{ \langle Tm \rangle \mid Tm \text{ accepts REL} \} = \text{Set of all TMs}$ ⇨ Decidable

By def every TM accepts REL

(37) $\{ \langle Tm \rangle \mid Tm \text{ accepts not REL} \} = \{ \}$


$L_1 = \{a^n b^n \mid n \geq 0\}$
$L_2 = \{a^n b^m \mid n \geq 0, m \geq 0\}$
$L_3 = \{\}$ (empty)

Can you design TM that accepts given Reg ?

↳ We can

can you verify given TM accepts regular ?

↳ Not REC

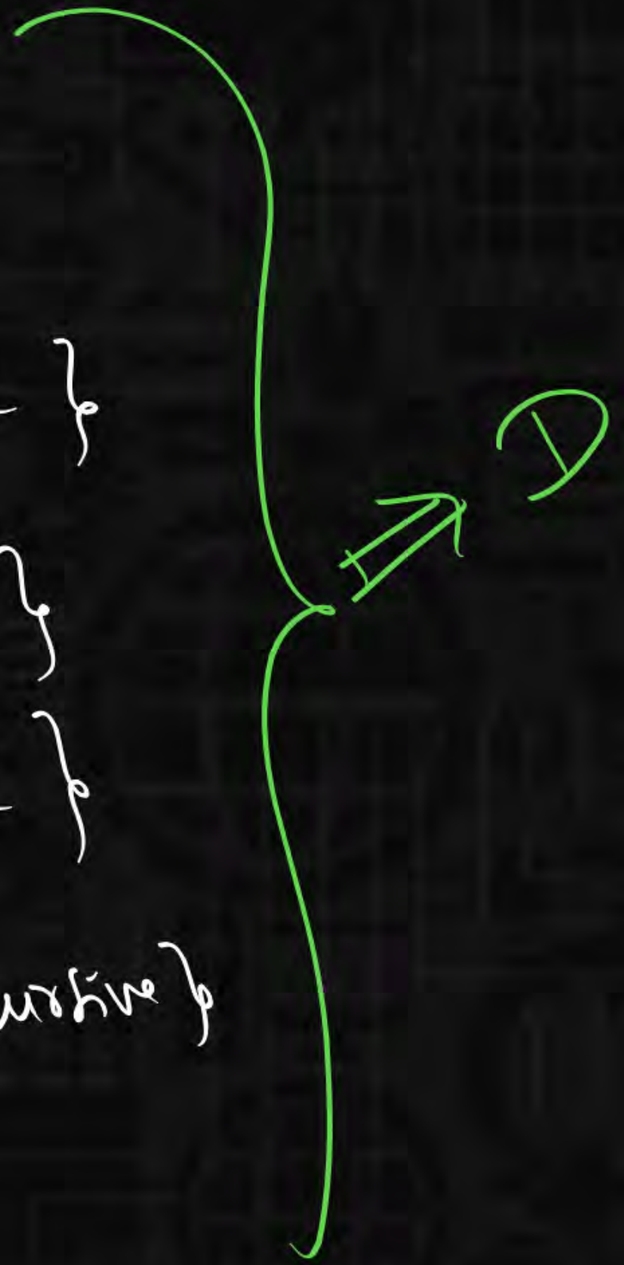
(38) { DFA | DFA accepts Regular }

(39) { DPDA | DPDA accepts DCFL }

(40) { PDA | PDA accepts CFL }

(41) { LBA | LBA accepts CSL }

(42) { Htm | Htm accepts Recursive }



(43) { TM | TM accepts abb } → SDUD (RE but not rec)

Membership

(44) { TM | TM accepts some 4 length string }

(45) { TM | TM accepts some 2014 length string }

(46) { Tm | Tm accepts all 3 length strings }

(47) { T_m | T_m accepts only abb }

(48) { T_m | T_m accepts only 4 length strings }

→ Not REL

(49) $\{ TM \mid$ Tm accepts ab within 5 steps $\} \Rightarrow$ Decidable

Whether Tm accepts ab within 5 moves

→ Yes: ab is accepted within 5 steps

0 1 2 3 4 5 } Halts at final

→ NO: ab is not accepted within 5 steps

TM definitely takes 6^{th} step

***

(50) { TM | TM accepts some string within ~~2023~~ 5 steps }

⇒ Decidable

abaaba

5 Yes No

Check upto ~~2023~~ 5 length strings

If no string halts at 5

TM

ε

Check upto 5 steps

a

b

5 length

(51) $\{ \langle M \rangle \mid M \text{ takes atleast } 3 \text{ steps on } w \}$ (≥ 3 steps) ⇒ Decidable

given w

↳ if M takes 3^{rd} step ⇒ Yes

if M halts in less than 3 steps ⇒ No

Yes: Every string takes 3rd step (upto 3 len)

No: Some string halts in less than 3 steps (upto 3 len)

52 ⇒ Decidable

{ TM | TM takes atleast 3 steps on all inputs }

For every input, TM should take min 3 steps

If we check upto 3 length strings then it means indirectly we checked all strings

$|w| = 0$
$|w| = 1$
$|w| = 2$
$|w| = 3$

Yes: Every string upto 3 length should take 4th step.

No: Some string upto 3 length halts within 3 steps.

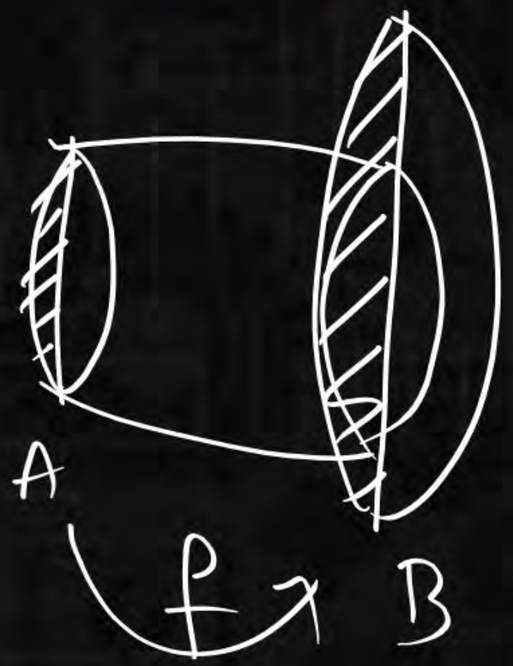
$\{ \langle M \rangle \mid M \text{ halts after } 3 \text{ steps} \}$

M never halts within 3 steps

⇓

we have to check upto 3 length strings

Reducibility :

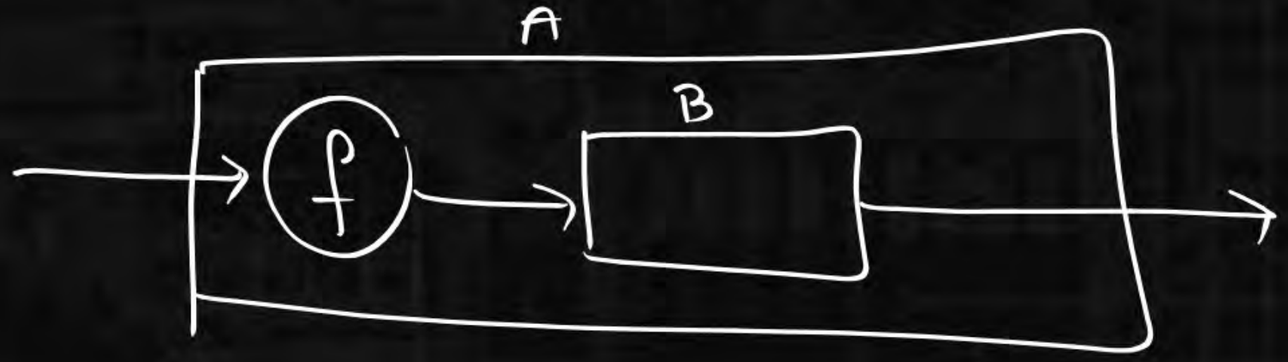


$A \leq_P B$

$A \propto_P B$

A is reducible to B

B is atleast as hard as 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 B is RE then A is RE

④ If A is not RE then B is not RE

Let $A \leq B$

(5) If A takes 100 days then B takes 100 or more (≥ 100)

(6) If B takes 100 days then A takes 100 or less (≤ 100)

(7) If B is RE then which of the following is TRUE?

a) A is Recursive

b) A is RE

c) A is not RE

d) A is RE but not Rec

Let $A \leq B$

(8) If B is RE then which of the following is possible?

a) A is Recursive
b) A is RE
c) A is not RE
d) A is RE but not Rec

Summary

Words ⇒ statement ⇒ meaning ⇒ logic

Extra class:

- countable Vs REL Vs Decidable
- Simplified CFG
- CNF & GNF
- CYK Algo
- GATE PYQs
- Doubts
- conversions FA ⇔ LLG/RLG, ...


THANK YOU GW
SOLDIERS !