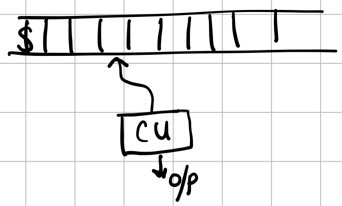
UNIT 4 :

– TM with a Stay option :

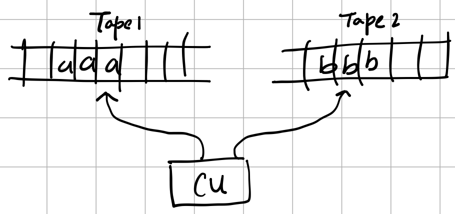
$\{L, R, S\}$

$\delta(q_i, a) = (q_j, x, S)$

– TM with Semi-∞ Tape :

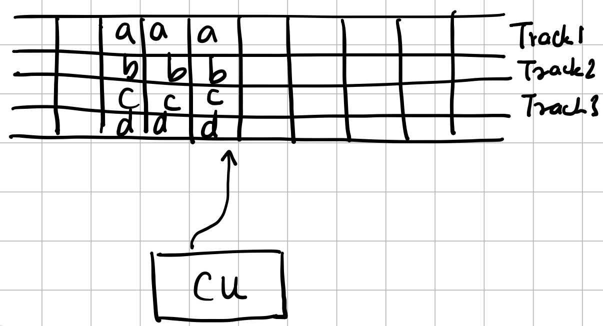


– TM with multiple tapes



$\delta(q_i, [a, b]) = (q_j, [x, y], [L, R])$

– TM with multiple tracks:



UNIVERSAL TURING MACHINES

2 inputs : → TM specification
→ Input string

o/p : YES | NO

Context Sensitive Grammar

$A \rightarrow xbc$

$xAy \rightarrow xbcy$

CFG: $G = (V, T, S, P)$

$A \rightarrow B$

$A \in V \qquad B \in (V \cup T)^*$

CSG: $G = (V, T, S, P)$

$A \rightarrow B$

① $A, B \in (V \cup T)^+$

removing restriction being put on LHS.

② $|A| \leq B$

$aA \rightarrow Aaa$

$A \rightarrow aaA$

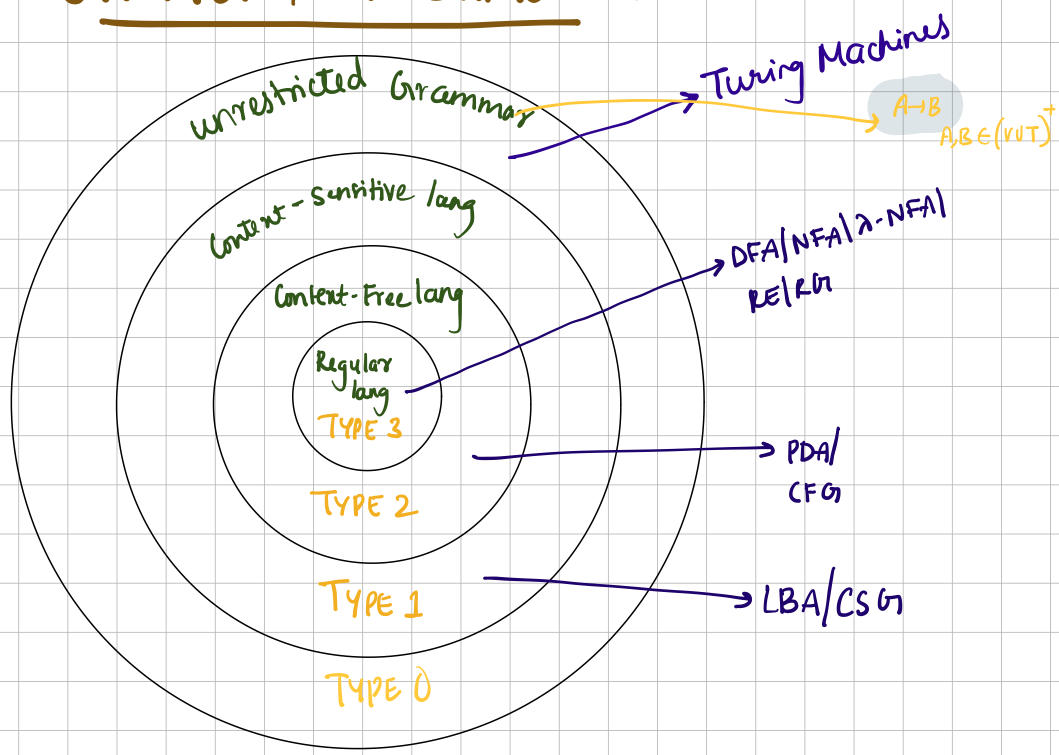
$AB \rightarrow aaa$

String substitution → never decrease

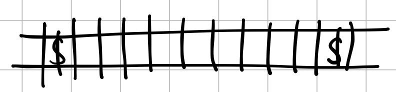
Stay the same (or) → Increase

$A \rightarrow aa$

CHOMSKY HIERARCHY



Linear Bounded Automata

① 

based on Input * Linear function

UnRestricted Grammar

$A \to B$

$A, B \in (V \cup T)^+$

Recursive language

* if invalid → goes to NF States
  ↓
  and clearly tells invalid

Recursively Enumerable language

* PCP
  you never know if it's invalid!
  ↓
  enters into an ∞ loop!

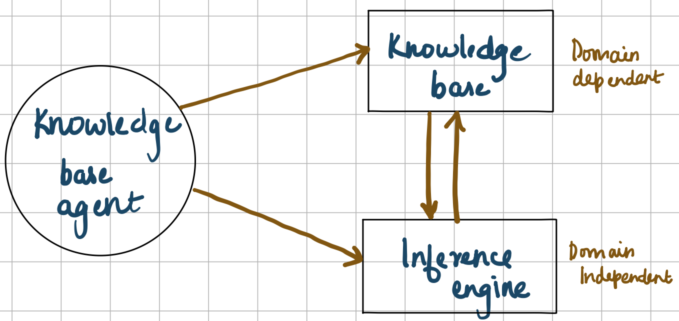
Tur

Turing Machine Halting Problem

→ recursively enumerable language

LOGIC

12/11/25



representing logic

- → Proposition logic
- → First Order logic

KB [AND]

$\neg B_{1,1}$ $\quad B_{1,2}$

$\neg S_{1,1}$ $\quad \neg G_{1,2}$

$\neg G_{1,1}$ $\quad \neg S_{1,2}$

$B_{1,2} \leftrightarrow P_{1,1} \vee P_{2,2} \vee P_{1,3}$

↓ $P_{1,3}$?

$B_{1,1}$	$\neg B_{1,1}$	$S_{1,1}$	$\neg S_{1,1}$	$G_{1,1}$	$\neg G_{1,1}$	$K_B = R_1 \wedge R_2 \wedge R_3 \ldots$

R1, R2, R3

Wumpus World

	1	2	3	4
4			Breeze	Pit
3	W Stench	Breeze Stench glitter	Pit	Breeze
2	Stench		Breeze	
1	Start	Breeze	Pit	Breeze

Proofs

* proof by enumeration.
* proof by resolution →

Inference rules

1. $(A \cap B) \equiv (B \wedge A)$
2. $(A \cup B) \equiv (B \cup A)$
3. $\neg(\neg A) \equiv A$
4. $A \wedge (B \wedge C) \equiv ((A \wedge B) \wedge C)$
5. $A \cup (B \vee C) \equiv (A \vee B) \vee C$
6. $A \rightarrow B \equiv \sim A \rightarrow \sim B$
7. $A \rightarrow B \equiv \sim A \cup B$
8. $A \leftrightarrow B \equiv (A \leftrightarrow B) \wedge (B \rightarrow A)$
9. $\sim(A \wedge B) \equiv \sim A \vee \sim B$
10. $\sim(A \cup B) \equiv \sim A \cap \sim B$
11. $A \cap (B \cup C) \equiv (A \cap B) \cup (A \cap C)$
12. $A \vee (B \wedge C) \equiv (A \cup B) (\wedge A \cup C)$

Proposition Logic

Declarative statements which are either T/F

Atomic Statement:
> one proposition

Complex Statement: → combination of propositions

* Conjunction operation / and ($\wedge$)
* Disjunction operation / or ($\vee$)
* Negation operation ($\sim$) / ($\neg$)
* Implier ($\rightarrow$)
* Bidirectional operation ($\leftrightarrow$)

Truth Table

P	Q	$P \wedge Q$ (AND)	$P \vee Q$ (OR)	$\sim P$ (NOT)	$P \to Q$ (IF-THEN)	$P \leftrightarrow Q$ (IFF)
T	T	T	T	F	T	T
T	F	F	T	F	F	F
F	T	F	T	T	T	F
F	F	F	F	T	T	T

IFF $\to (P \to Q) \wedge (Q \to P)$

> Tautology : always True

> Contradiction : always False

eg: $(P \wedge (Q \to R)) \to P \implies$ Tautology!

P	Q	R	$Q \to R$	$P \wedge (Q \to R)$	$[P \wedge (Q \to R)] \to P$
T	T	T	T	T	T
T	T	F	F	F	T
T	F	T	T	T	T
T	F	F	T	T	T
F	T	T	T	F	T
F	T	F	F	F	T
F	F	T	T	F	T
F	F	F	T	F	T

eg 2: $(P \vee \sim P) \to \sim(P \vee \sim P) \implies$ Contradiction!

P	$\sim P$	$(P \vee \sim P)$	$\sim(P \vee \sim P)$	$(P \vee \sim P) \to \sim(P \vee \sim P)$
T	F	T	F	F
F	T	T	F	F

Terms

* Equivalence $\to (\equiv)$
* Validity $\to$ (Tautology)
* Satisfyability $\to$ Some T / Some F

①

1. If it rains, Joy will bring his umbrella.
2. If Joy has an umbrella, he doesn't get wet.
3. If it doesn't rain, Joy doesn't get wet

→ Prove that Joy doesn't get wet?

P : it rains

Q : Joy → brings umbrella

R : Joy will get wet

*Prove $\sim R \rightarrow$ true

$R_1 : P \rightarrow Q$

$R2 : Q \rightarrow \sim R$

$R3 : \sim P \rightarrow \sim R$

Proof By Resolution

① Convert everything into and of ors

$R1 : \sim P \cup Q$ [rule 7]

$R2 : \sim Q \vee \sim R$

$R3 : P \cup \sim R$

② New Rule — Inverse of proof

$R_4 : \sim(\sim R) : R$

③ R_5 : $\boxed{\sim P \cup Q}$ (R1), $\boxed{P \cup \sim R}$ (R3) $\rightarrow \boxed{Q \vee \sim R}$ R5

④ $\boxed{\sim P \cup Q}$, $\boxed{\sim Q \cup \sim R}$ $\rightarrow \boxed{\sim P \cup \sim R}$ R6

$R_5 - R_2 \rightarrow \sim R$ R_7

$R_4 - R_7 \rightarrow R \cdot \sim R = R_8 = \emptyset$

②

1. If either Python or C prog is required, the all Stud take CS
2. Python and C are required

Prove that all Student take CS.

P : Python req

Q : C req

R : All students will take CS

$R1 : (P \vee Q) \rightarrow R$

$R2 : P \wedge Q$

To prove : R

Proof by Resolution

① $R_1 : \sim(P \cup Q) \cup R$ (∨ ∩ ∨ ∩ (∨)

$(\sim P \cap \sim Q) \cup R$

$(\sim P \cup R) \cap (\sim Q \cup R)$ (and of ors)

$R_{11} : \sim P \cup R$

$R_{12} : \sim Q \cup R$

$R2 : P \cap Q$

$R_{21} : P$

$R_{22} : Q$

$R_3 : \sim R$

$(R_{11}$ & $R_{21})$ → If and, chop separate

$\Downarrow$

$R_4 : R$

$(R_4$ & $R_3)$

$\Downarrow$

$R_5 : \emptyset$

③

1 : If there is cream, I'll drink coffee

2 : If there is donut, I'll drink coffee

3. There's no cream, there is a donut

Infer the conclusion : I drink coffee

P : There is cream

Q : There is donut

R : I drink coffee

$R1 : P \rightarrow R$

$R2 : Q \rightarrow R$

$R3 : \sim P \cap Q$

Prove R

① $R1 : \sim P \cup R$

$R2 : \sim Q \cup R$

$R3 : \sim P \wedge Q$

$R_{31} : \sim P$

$R_{32} : Q$

$R4 : \sim R$

$\boxed{R_2, R_{32}}$

$\Downarrow$

$R_5 : R$

$\boxed{R_4, R_5}$

$R_6 : \emptyset$

			R1	R2		R3	$R_1 \cap R_2 \cap R_3$
P	Q	R	$P \rightarrow R$	$Q \rightarrow R$	$\sim P$	$\sim P \wedge Q$	KB
T	T	T	T	T	F	F	F
T	T	F	F	F	F	F	F
T	F	T	T	T	F	F	F
T	F	F	F	T	F	F	F
F	T	T	T	T	T	T	T
F	T	F	T	F	T	T	F
F	F	T	T	T	T	F	F
F	F	F	T	T	T	F	F

$R \Rightarrow T \Longrightarrow$ I drink coffee ✓

19/11/25

FIRST ORDER LOGIC (FOL)

$\forall_x \rightarrow$ for all x

$\exists_x \rightarrow$ There exists

① Ram likes Mango

likes(Ram, Mango)

② Emma in a girl

girl(Emma)

③ Bhaskar likes food

likes(Bhaskar, food)

④ Pluto is a man

man(Pluto)

⑤ Priya owns gold

owns(Priya, gold)

⑥ Harry is taller than Jacob

Taller(Harry, Jacob)

⑦ Kiwi is a fruit

fruit(Kiwi)

⑧ Kevin is son of John

Son(Kevin, John)

⑨ All kings are persons

$\forall_x \ King(x) \longrightarrow person(x)$

⑩ Nobody loves John

$\forall_x \sim loves(x, John)$

⑪ Everyone has a friend

$\forall_x \forall_y (friend(x,y))$ X

$\exists_x \forall_y (friend(x,y))$

⑫ All people who are not poor and smart are happy

$\forall_x ((\sim poor(x) \wedge smart(x)) \longrightarrow happy(x))$

⑬ All Romans are either loyal to caeser or hate him

$\forall_x ((roman(x) \longrightarrow (loyal(x, caeser) \vee hate(x, caeser)))$

⑭ everyone is loyal to someone

$\forall_x \exists_y \ loyal(x,y)$

⑮ Rani's father is Raja's father

$father(father(rani), Raja)$

- Harry, Ron and Draco are students of Hogwart school of Wizards
- Every student is either wicked or good Quiditch player or both
- No Quiditch player likes rain and all wicked students like potions.
- Draco dislikes whatever harry likes and likes whatever harry dislikes.
- Draco likes rain and potions.

Can we conclude, there exists good in quiditch but not in potions.

R_1: Student (Harry)

R_2: Student (Ron)

R_3: Student (Draco)

R_4: $\sim$Student(x) $\vee$ (wicked(x) $\vee$ quiditch(x))

R_5: $\forall_x$ $\sim$Quiditch(x) $\vee$ $\sim$likes$(x$, Rain$)$

R_6: $\forall_x$ $\sim$Wicked(x) $\vee$ likes $(x$, Potion$)$

R_7: $\forall_x$ $\sim$likes(Harry, x) $\vee$ $\sim$likes(Draco, x)

R_8: $\forall_x$ likes(Draco, x) $\vee$ likes(harry, x)

R_9: likes (Draco, rain)

R_{10}: likes (Draco, potion)

② $\forall_x$ Student(x) $\longrightarrow$ (wicked(x) $\vee$ good quiditch(x))

$P \rightarrow Q = \sim P \vee Q$

③ $\sim$like (quid, rain)

$\forall_x$ $Q(x) \rightarrow \sim$likes$(x$, rain$)$

$P \rightarrow Q$

$\sim P \vee Q$

$\rightarrow$ $\forall_x$ wicked(x) = likes$(x$, potions$)$
P Q

wicked(x) $\rightarrow$ likes$(x$, potions$)$
P Q

$\sim$ wicked(x) $\vee$ likes$(x$, potions$)$

④

$\forall_x$ likes(Harry, x) $\leftrightarrow$ $\sim$likes(Draco, x)
P Q

$P \leftrightarrow Q$

$(P \rightarrow Q) \wedge (Q \rightarrow P)$

likes$(x$, potions$)$ $\rightarrow$ wicked (x)
Q P

$\sim$likes (x,p) $\vee$ wicked(x)

Q : $\exists_x$ Quiditch(x) $\wedge$ $\sim$likes$(x$, potion$)$

R_{11} : neg $\rightarrow$ $\forall_x$ $\sim$Quiditch(x) $\vee$ likes $(x$, potion$)$

R_7 & R_8 $\rightarrow$ ϕ //

Q)

1: Cakes are delicious

2: Pickles are delicious

3: Biryani is delicious

4: Pickles are spicy

5: Priya relishes coffee

6: Prakash relishes coffee

7: Priya likes foods that are delicious

8: Prakash likes food which are spicy & delicious

9: Priya is fond of driving

10: Prakash is fond of cartoons

Represent these knowledge as FOL statements in PROLOGUE language.

LOGIC PROG LANGUAGE

LISP   PROLOG

1. Delicious (cakes).
2. Delicious (pickles).
3. Delicious (Biryani).
4. Spicy (pickles).
5. relishes (priya, coffee).
6. relishes (prakash, coffee).
7. likes (priya, Food) :- Delicious (Food)
8. likes (prakash, food) :- spicy(food) , delicious (food)

   ($\longleftrightarrow$)   (&)
9. fond (priya, driving)
10. fond (Prakash, cartoons)

Swish → online prolog compiler

→ → :-
∧ → ,
∨ → ;

who relishes coffee & fond of driving

↳ relishes (X, coffee), fond (X, driving).

↳ X = priya