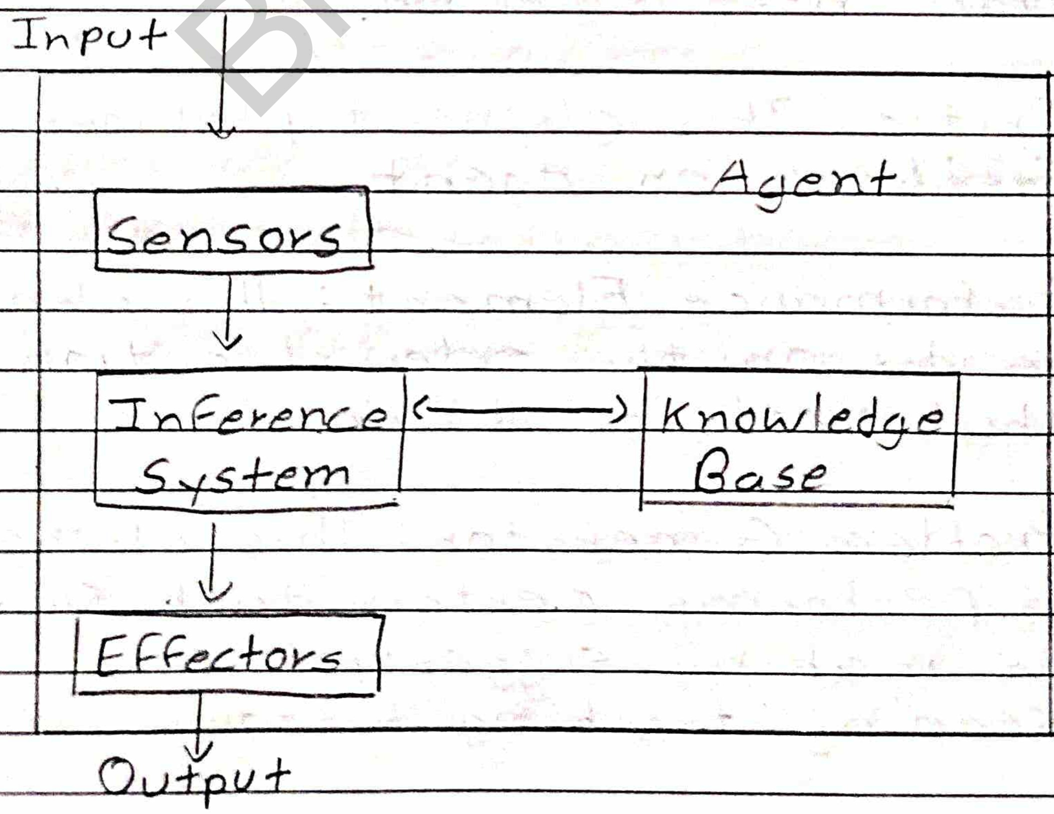
Knowledge and Reasoning

* Explain Knowledge Based Agent in AI.

=> A Knowledge Based Agent in AI is an intelligent agent that operates based on a knowledge base.

This Type of agent needs a knowledge about the real world for making or taking decisions.

Environment



This agent has capability of maintaining an internal state of knowledge, reason over that Knowledge and update their knowledge after take action.

This type of agent can take action using knowledge and update their action according to their observations.

Knowledge - Based Agent are divided into two parts.

(i) Knowledge - Base
(ii) Inference System

(i) Knowledge - Base:

The knowledge - base system is used to stored information in Knowledge - based system.

Knowledge - Base can be represented knowledge using various formalisms such as rules, frames, ontologies, semantic network etc.

The Knowledge Base is typically

created and maintained by the Knowledge engineers.

(ii) Inference System:

Inference System or Engine is a knowledge-based system to infer new things at "knowledge level".

It uses various inference mechanism and algorithm to perform reasoning and decision-making based on the available knowledge.

The output of the inference system can be used for tasks such as problem-solving or generating responses to queries.

Inference system works mainly on two rules.

(a) Forward Chaining
(b) Backward Chaining

* Explain Techniques of Knowledge Representation.

=> There are Main Four ways to Represent knowledge.

(a) Logical Representation
(b) Semantic Network Representation
(c) Frame Representation
(d) Production Rules

(a) Logical Representation:

In Logical Representation, knowledge is represent using the Formal logic, such as predicate or propositional logic.

Knowledge statements are represent using logical symbols, predicates, variables and quantifiers.

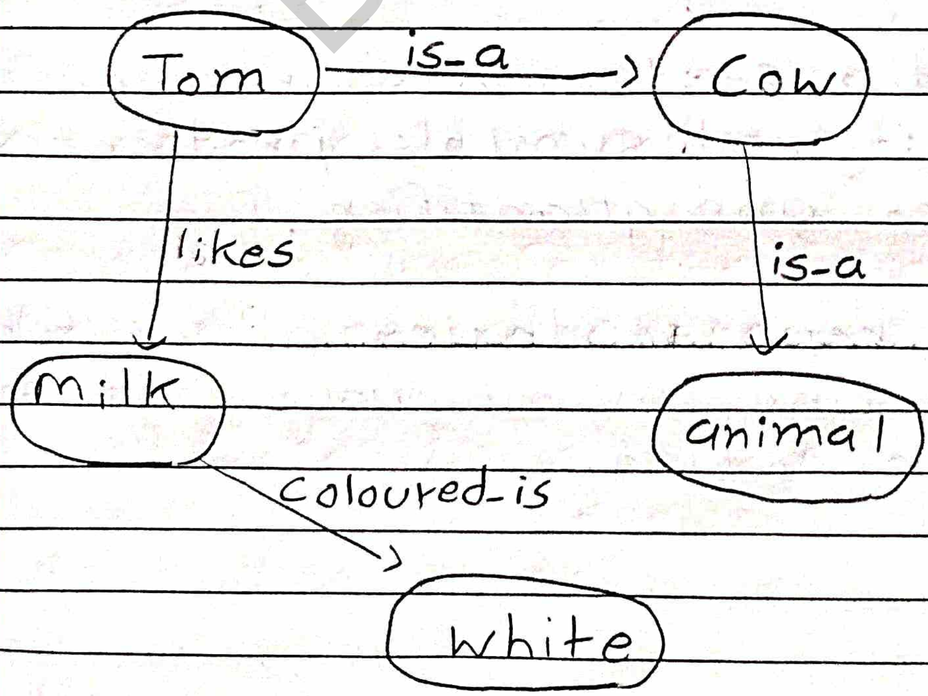
Logical representation allows for precise and unambiguous statements representation.

(b) Semantic Network Representation:

Semantic networks represent knowledge using a network of an interconnected nodes.

In Semantic network, Nodes are represent entities and edges are represent relationships between them.

Ex. Tom is a cow.
Cow is a animal
Tom likes milk
milk coloured is white.



(c) Frame Representation:

Frame Representation organizes knowledge into structured Frames or templates.

Frames consist of slots or attributes that describe properties, Features and relationships of the object.

Ex. Tom is an engineer as a profession and his age is 27 and lives in city Gandhinagar and likes Drawing and Swimming.

Slots	Filter
Name	Tom
Profession	Engineer
Age	27
Address	Gandhinagar
Hobbies	Drawing, Swimming

(d) Production Rules:

Production rules represent knowledge as a set of conditional rules or patterns in the form of "if-then" statements.

Each rule consists of conditions that trigger action when the condition is satisfied.

* Explain First Order Logic in AI.

=> First Order Logic also known as predicate logic or first-order predicate.

First Order Logic is a formal language used in AI and logic-based systems to represent knowledge.

FOL allows for the representation of complex knowledge structures.

FOL is allows for the formal semantics of logical expressions.

First Order Logic is a powerful language that develops information about the object.

Basic Elements of First Order Logic :

1 Variables : Used to represent placeholders for Objects
Ex. X, Y, Z etc.

2 Constants : Used to represent specific objects.
Ex. John, Tom, a etc.

3 Predicates : Used to represent relations that can be true or false for objects.
Ex. (P(Q), X(A, B)) etc.

4 Functions : Used to represent mapping from objects to other objects.
Ex. (F(x), g(x, y))

5 Quantifiers : It includes Existential (∃) and universal quantifiers.

FOL is used in knowledge representation and reasoning systems.

* Explain Unification Algorithm.

=> Unification in AI refers to the process of finding a substitution that makes two logical or equivalent expressions.

Unification used to finding substitution for variables in two logical expressions to make them identical.

The unification algorithm systematically compares and matches elements of two expressions.

=> Conditions of Unification :

1 Predicate symbol must be same, atoms or expression with different predicate symbol.

2 Number of arguments in both expression must be identical.

3 If there are two similar variable in same expression then unification algorithm is fail

=) Unification Algorithm:

Unify(Ψ_1, Ψ_2)

1 If Ψ_1, or Ψ_2 is a variable or constant then,

(a) If Ψ_1 or Ψ_2 are identical then return nil.

(b) else if Ψ_1 is a variable,
(i) then if Ψ_1 occurs in Ψ_2 then return Fail.
(ii) else return $\{(\Psi_2 / \Psi_1)\}$.

(c) else if Ψ_2 is a variable,
(i) If Ψ_2 occurs in Ψ_1 then return Fail.
(ii) else return $\{(\Psi_1 / \Psi_2)\}$.

(d) else return Fail.

2 If the initial Predicate symbol in Ψ_1 and Ψ_2 are not same, then return Fail.

3 If Ψ_1 and Ψ_2 have a different number of arguments, then return Fail.

4 Set substitution set(SUBST) to nil.

5 For i = 1 to the number of elements in Ψ_1.

(a) Call Unify function with the ith element of Ψ_1 and ith element of Ψ_2 and put the result into S.

(b) If S = fail then return fail

(c) If S ≠ nil then do,
   (i) Apply S to the reminder of both L1 and L2.
   (ii) SUBST = APPEND (S, SUBST).

6 Return SUBST.

* Reasoning in Artificial Intelligence.

=> Reasoning is a way to infer facts from existing data.

There are Six Types of a Reasoning.

SMVS

Page No.

Date : / /

1 Deductive Reasoning:

Deductive Reasoning is used when conclusions are drawn logically from given axioms.

If the Axioms are true then rule of inference are vaild.

Ex. All Humans are Mortal.
Tom is a Human
Therefore,
Tom is Mortal.

2 Inductive Reasoning:

Inductive Reasoning involves generalizing conclusions from specific observations.

It is based on probabilities and patterns rather than strict logical inference

Ex. All observed swans are white.
Therefore,
All swans are probably white.

3 Abductive Reasoning:

Abductive Reasoning is about forming hypotheses to explain data.

It involves generating the best explanation given the available evidence.

Ex. You find a wet umbrella by the door

Hypothesis - Could be it's raining outside.

4 Common Sense Reasoning:

Common Sense Reasoning has ability of AI Systems to make decisions based on general knowledge and everyday understanding.

Ex. If it's hot outside, people are likely to wear lighter clothes.

5 Monotic Reasoning:

Monotic Reasoning is a type of

reasoning where adding more knowledge leads to made the conclusions.

Ex. If it's raining, then the ground is wet.

6 Non-monotonic Reasoning:

Non-monotonic Reasoning is the opposite of monotonic reasoning.

It allows for conclusions to be modified based on new information.

Ex. By default, Birds can fly. However if we learn about a bird that can not fly (Ex. ostrich).

* Explain Difference Between Procedural and Declarative knowledge.

	Procedral Knowledge	Declarative Knowledge
1	Also known as Interpretive knowledge.	Also known as Descriptive knowledge.
2	Procedral knowledge is not more popular.	Declarative knowledge is more popular.
3	It can't be easily communicate.	It can be easily communicate.
4	It is process oriented in the nature.	It is data oriented in the nature.
5	In this, debugging and validation is not easy.	In this debgging and validation is easy.
6	It is less effective in competitive programming	It is more effective in commpetitive programming.

Page No.
Date : / /

* Explain Resolution with example.

=> Resolution is used to represent knowledge to derive new information from existing information.

Steps for Resolution :

1 Convert facts into the First Order Logic.

2 Convert First Order Logic to CNF

(1) Eliminate Implication

$\alpha \rightarrow \beta = \neg \alpha \vee \beta$

$\alpha \leftrightarrow \beta = (\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)$

(2) Standardize Variable : All the used variable should be different.

(3) Move Negation Inwards

- $\neg(\forall x\ P(x)) = \exists x\ \neg P(x)$
- $\neg(\exists x\ P(x)) = \forall x\ \neg P(x)$
- $\neg(\alpha \vee \beta) = \neg\alpha \wedge \neg\beta$
- $\neg(\alpha \wedge \beta) = \neg\alpha \vee \neg\beta$
- $\neg(\neg\alpha) = \alpha$

Page No.
Date : / /

(4) Remove Existential quantifier ∃.

(5) Remove Universal Quantifier.

3 Negate the statement which needs to prove (by ~~com~~ contradiction)

4 Draw Resolution Graph

Ex. - All People who are graduating are Happy
- All Happy people smile.
- Someone is smile -> Prove that
- Someone is Graduating.

=>

Step 1 : Convert into FOL.

- $\forall_x (Graduating(x) \rightarrow happy(x))$

- $\exists_x \, Graduating(x)$

- $\forall_x (happy(x) \Rightarrow smile(x))$

Prove that, $\exists_x \, smile(x)$

- Step 2 : Convert into CNF.

(1) Eliminate Implication

- $\forall_x (\neg Graduating(x) \vee Happy(x))$.

- $\forall_x (\neg Happy(x) \vee smile(x))$

- $\exists_x$ Graduating(x)

- $\exists_x$ Smile(x)

(2) Standardize Variable.

- $\forall_x (\neg Graduating(x) \vee Happy(x))$.

- $\forall_y (\neg Happy(y) \vee smile(y))$

- $\exists_z$ Graduating(z)

- $\neg(\exists_w$ Smile(w)) → Prove statemen

(3) Move Negation Inwards

Same as upper case.

$\forall_w \neg Smile(w)$

(4) Remove ∃ quantifier.

Graduating(A)

(5) Drop ∀ quantifier

- ¬Graduating(x) ∨ Happy(x)

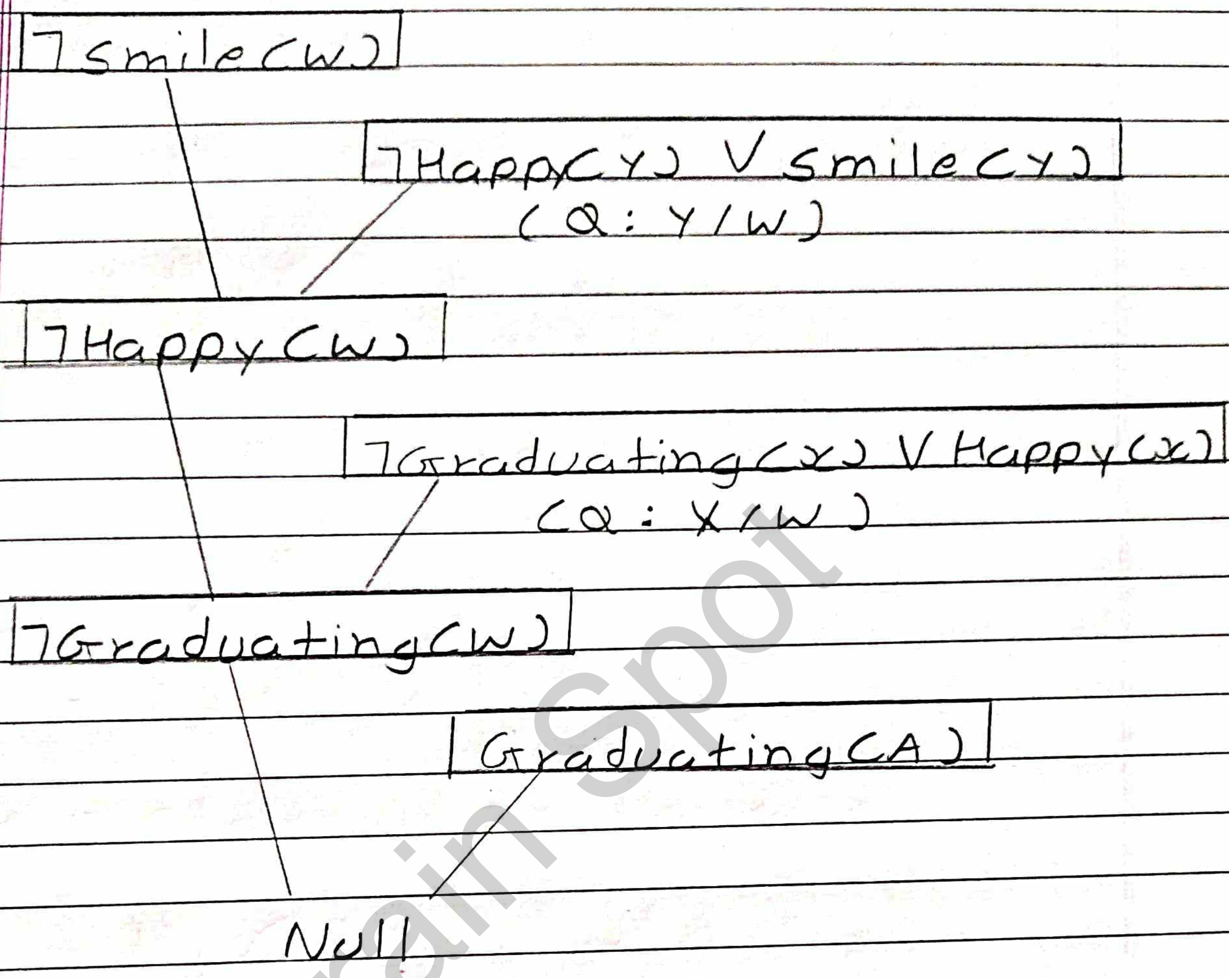
- ¬Happy(y) ∨ smile(y)

- Graduating(A)

- ¬smile(w)

3 Use contradication for prove statement
- ¬smile(w).

4 Tree (Resolution).



Hence, Prove Someone is smiling.