LAB-10
PRELAB

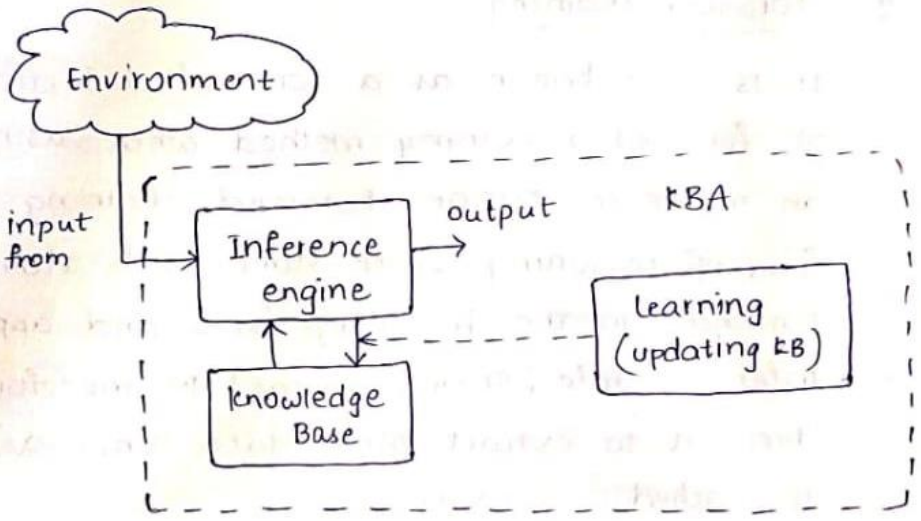
1. Write a short note on Knowledge Based agent with its architecture and write the two functions of Knowledge Based agent. Write a simple algorithm on its functionality.

AI Lab-10

19003187
Radhakrishna

pre-lab

1. Knowledge based agents are those agents who have the capability of maintaining an internal stage of knowledge, reason over that knowledge, update their knowledge after observations and take actions. These agents can represent the world with some formal representation and act intelligently.



Functions :-

An agent should be able to represent states, actions etc.

An agent should be able to incorporate new percepts.

Algorithm :-

```
function KB-Agent (percept) returns an
                                    action

static KB, a knowledge-base t,
       a counter, intially 0, indicating time
TELL (KB, Make-percept_sentence (percept, t))
action ← Ask (KB, Make-Action-Query (t))
TELL (KB, Make-Action-sentence (action, t))
t ← t + 1
return Action
```

2. State and explain about forward chaining with an example.

2. Forward chaining

It is also known as a forward deduction or forward reasoning method when using an inference engine. Forward chaining is a form of reasoning which start with atomic sentences in the knowledge base and applies inference rule (Modus ponens) in the forward direction to extract more data until a goal is reached.

The forward chaining algorithm starts from known facts, triggers all rules whose premises are satisfied, and add their conclusion to known facts. This process repeats until the problem is solved.

properties of Forward chaining :-

→ It is a down up approach, as it moves from bottom to top

→ It is a process of making a conclusion based on known facts or data, by starting from initial state and reaches the goal state

→ Forward chaining approach is also called as data-driven as we reach to the goal using available data.

→ Forward chaining approach is commonly used in the expert system, such as cups, business and production rule systems.

consider the following famous example which we will use in both approaches

Example :-

" As per the law, it is a crime for an American to sell weapons to hostile nations. country A, an enemy of America, has some missiles, and all the missiles were sold to it by Robert who is an American citizen.

prove that "Robert is criminal"

To solve the above problem first, we will convert all the above facts into first order definite clauses and then we will use a forward chaining algorithm to reach the goal.

IN LAB

1. Write a python code for the following inference rules and facts such that the inference engine generates a list. Implement the code using Forward Chaining.

Seed(A) ==> Plant(A).
Plant(A) ==> Fruit(A).
Plant(A),Eating(A) ==> Human(A).
Plant("Mango").
Eating("Mango").
Seed("Sprouts").

```
global facts
global is_changed

is_changed = True
facts =[["plants","mango"],["eating","mango"],["seed","sprouts"]]

def assert_fact(fact):
    global facts
    global is_changed
    if not fact in facts:
        facts += [fact]
        is_changed = True

while is_changed:
    is_changed = False
    for A1 in facts:
        if A1[0] == "seed":
            assert_fact(["plant",A1[1]])
        if A1[0] == "plant":
            assert_fact(["fruit",A1[1]])
        if A1[0] == "plant" and ["eating",A1[1]] in facts:
            assert_fact(["human",A1[1]])

print(facts)
```

POST LAB

1. Translating English into first order logic
    1. Every gal in Constantinople lives in Istanbul, not Constantinople.
    2. Every new beginning comes from some other beginning end.

postlab

1. 1. $\forall x \left[Gal(x) \wedge In(constanlinople) \right] \rightarrow$

$\left[lives(x, Istanbul) \wedge \neg lives(x, constan-tinople) \right]$

2. we need to predicate beginning that tests it something is a beginning, a predicate comes from and a function end

$\forall x \; \exists y \; Beginning(x) \rightarrow [Beginning(y) \wedge comes\ from(x, end(y))]$

2 a) Apply backward chaining and prove that Gita loves Kurtis.

Ans) Convert the given axioms into FOPL as:

x: clothes(x)→loves(Gita, x).

Suits(x)→Clothes(x).

Jackets(x)→Clothes(x).

wears(x,y)→Λ ¬bad(y)→Clothes(x)

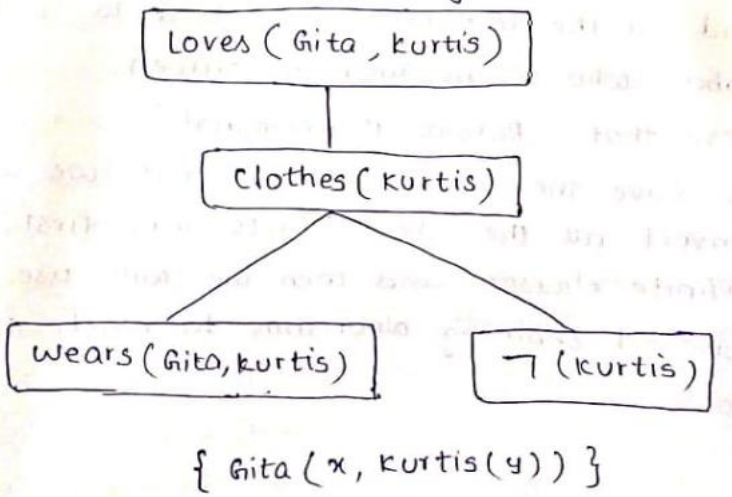
wears(Sita,skirt)Λ good(Sita)

wears(Sita,x)→wears(Renu,x)

2a. To prove: Gita loves kurtis

FOPL: loves (Gita, kurtis)

Apply Backward chaining:



{ Gita (x, kurtis (y)) }

b) Derive forward chaining using the given known facts to prove Tony is blue.

- Tony barks.
- Tony eats bone.

2b. On replacing D with Tony in (3), it becomes if Tony is a dog, then Tony is blue.

Thus, the goal is matched with above axiom

Now, we have to prove Tony is a dog (new goal)

If tony barks and tony eats bone, then tony is a dog - (new goal)

Again, the goal is achieved.

Now, we have to prove Tony barks and Tony eats bone - (new goal)

As we can see, the goal is a combination of two sentences which can further divided as:

Tony barks

Tony eats bone

From (1), it is clear that Tony is a dog.

Hence, Tony is blue.