**Batch: B1 Roll No.: 16010122221**

**Experiment / assignment / tutorial No. 3**

**Grade: AA / AB / BB / BC / CC / CD /DD**

**Signature of the Staff In-charge with date**

|  |
| --- |
| **Title:** Implementation of Goal based agent architecture using PROLOG. |

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Objective:** To use the concepts of knowledge engineering to design and solve moderate complex problem.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Expected Outcome of Experiment:**

|  |  |
| --- | --- |
| **Course Outcome** | **After successful completion of the course students should be able to** |
| **CO1** | Design AI solution with appropriate choice of agent architecture |
| **CO3** | Represent and formulate the knowledge to solve the problems using various reasoning techniques |

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Books/ Journals/ Websites referred:**

1. **https://www.csupomona.edu/~jrfisher/www/prolog\_tutorial/contents.html**
2. **http://www.csupomona.edu/~jrfisher/www/prolog\_tutorial/pt\_framer.html**
3. **http://www.doc.gold.ac.uk/~mas02gw/prolog\_tutorial/prologpages/**
4. **http://classes.soe.ucsc.edu/cmps112/Spring03/languages/prolog/PrologIntro.pdf**
5. **“Prolog: Programming for Artificial Intelligence” by Ivan Bratko, Pearson education Publications**
6. **“Artificial Intelligence: a Modern Approach” by Russel and Norving, Pearson education Publications**
7. **“Artificial Intelligence” By Rich and knight, Tata Mcgraw Hill Publications**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Pre Lab/ Prior Concepts:**

Agents, Agent Architecture, Programming with PROLOG

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Historical Profile:**

Knowledge is vast, uncertain and continuously changing. These properties of knowledge make it difficult to arrive at a result. A murder mystery is a kind of situation which depicts the uncertain nature of knowledge and also emphasizes the need of choosing right clauses from entire knowledgebase to make a decision. He goal based agent architecture and some knowledge engineering can help in solutioning of such problems.

The logical agents are complex but they can reason and learn from the actions and new precepts. They are less like acting and think like humans but more like acting and thinking rational agents.

Knowledge and reasoning play a crucial role in dealing with partially observable environments. A knowledge based agent can combine the general knowledge with current percept to infer the hidden aspects of the current state prior to selecting actions.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**New Concepts to be learned:**

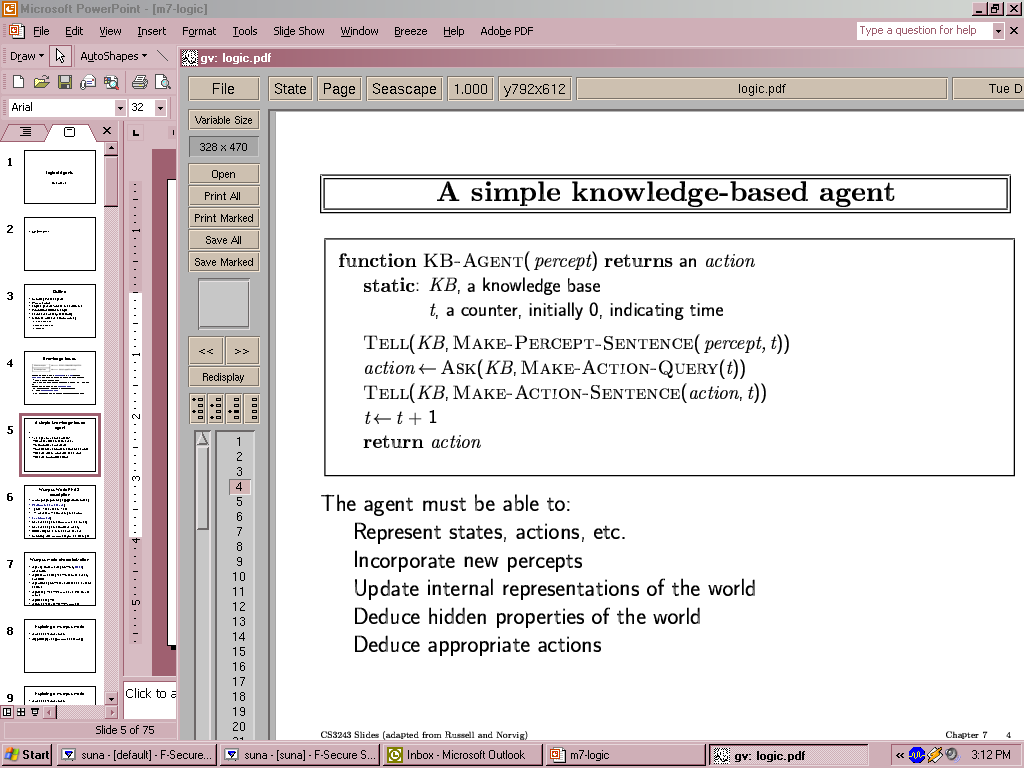
Knowledge engineering, implementing complex agent architecture, uncertainty in knowledge.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**The Knowledge Engineering Process**

1. Identify the task
2. Assemble the relevant knowledge
3. Decide on vocabulary of predicates, functions and constants
4. Encode general knowledge about the domain
5. Encode description of specific problem instance
6. Pose queries to the inference procedure and get answers
7. Debug the knowledge base

**Algorithm for KB-Agent:**



**Problem Statement:**

In Mumbai, wealthy businessman Rajesh Sharma (55) was found dead in his home, killed by poison. Detective Verma is looking at four suspects: Dr. Priya Sharma (his wife), Arjun Sharma (his son), Neha Kapoor (his niece), and Ramu (the family servant). Priya could inherit a lot of money, Arjun had been fighting with his father, Neha had feelings for Rajesh that he didn't return, and Ramu needed money badly. Priya says she was at the hospital and Neha was in court when Rajesh died. Both Arjun and Ramu were at home during that time. Tests showed that Priya, Arjun, and Ramu could get the poison, but Neha couldn't. During questioning, Priya stayed calm, Arjun seemed nervous, Neha was worried but helpful, and Ramu was very anxious. The detective needs to figure out who had the reason, chance, and ability to kill Rajesh.

**Knowledge Engineering steps applied to chosen problem:**

1. Problem Identification:

Created a murder mystery where businessman Rajesh was killed with poison, with multiple suspects and evidence.

2. Knowledge Acquisition:

Gathered facts about:

- People involved (age, gender, profession)

- Relationships between people

- Motives for each suspect

- Alibis during the time of murder

- Access to the murder weapon

- Behavior during questioning

3. Knowledge Representation:

Used predicates to represent:

- Basic facts about people: `person(Name, Age, Gender, Occupation)`

- Suspects: `suspect(Person)`

- Murder weapon: `killed\_with(Victim, Weapon)`

- Motives: `motive(Person, Reason)`

- Alibis: `alibi(Person, Location)`

- Relationships: `relation(Person1, Person2, RelationType)`

- Weapon access: `weapon\_access(Person, Access)`

- Behavior: `behavior(Person, BehaviorType)`

4. Rule Development:

Created rules to determine:

- Strong suspects: those with motive and access to weapon

- Primary culprit: strong suspect at home showing anxiety

- Accomplice: strong suspect at home showing nervousness

- Innocent people: those who aren't strong suspects

- Falsely suspected: suspects who aren't the culprit or accomplice

5. Inference Process:

The system uses backward chaining to:

- Determine who had means (weapon access)

- Determine who had a motive (inheritance, money, etc.)

- Determine who had opportunity (being at home)

- Consider behavior as additional evidence

- Combine these factors to identify culprit and accomplice

6. Testing and Refinement:

Created queries to test the system:

- Who are the strong suspects?

- Who is the primary culprit?

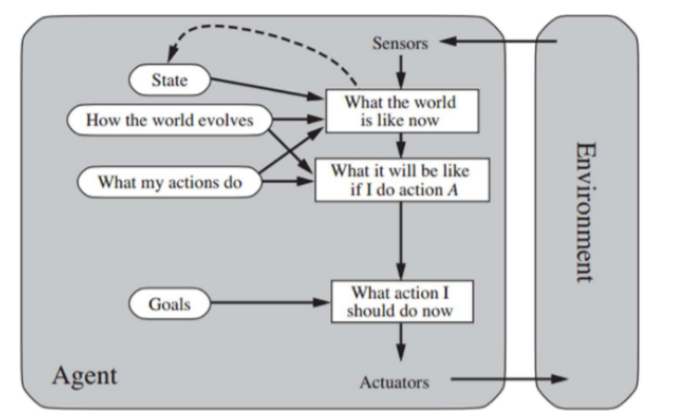
- Who might be an accomplice?

- Who is innocent?

7. Implementation:

Used Prolog as the logic programming language to implement the knowledge base and rules for inferencing.

**Agent Architecture** *(Justify the blocks)***:**



### What is the world like now?

This particular question focuses on understanding the current environment and circumstances. Rajesh Sharma, a wealthy businessman, has been found dead in his home, poisoned. There are four suspects: his wife, Dr. Priya Sharma; his son, Arjun Sharma; his niece, Neha Kapoor; and the family servant, Ramu. Each of them has a potential motive—inheritance, conflict, unreciprocated feelings, or financial desperation. Some claim alibis, while others were present at the crime scene. The goal is to uncover who among them had the motive, opportunity, and means to commit the murder.

### What will it be like if I do a particular action?

This block explores the outcome of taking a specific action in the current state. For example, verifying Priya’s alibi at the hospital or Neha’s claim of being in court could eliminate them as suspects. Investigating Arjun and Ramu’s actions at home during the time of the murder could provide clarity on their opportunity to commit the crime. Additionally, analyzing behavior during questioning—such as nervousness, calmness, or anxiety—might offer hints about guilt or innocence. Each action provides more information to refine the suspect list.

### What should I do now?

This block determines the next logical step based on gathered evidence and perceptions. The system can prioritize verifying alibis, checking access to the poison, and evaluating behavioral patterns. If Priya and Neha’s alibis hold, the investigation shifts to Arjun and Ramu. Based on the combined analysis of motive, opportunity, and behavior, the AI identifies the most likely culprit and any potential accomplices. This final decision is communicated back to the environment for further actions or resolution of the case.

**Code:-**

**person(rajesh, 55, m, businessman).**

**person(priya, 45, f, doctor).**

**person(arjun, 25, m, engineer).**

**person(neha, 30, f, lawyer).**

**person(ramu, 40, m, servant).**

**suspect(priya).**

**suspect(arjun).**

**suspect(neha).**

**suspect(ramu).**

**killed\_with(rajesh, poison).**

**motive(priya, inheritance).**

**motive(arjun, family\_dispute).**

**motive(neha, rejected\_affair).**

**motive(ramu, money).**

**alibi(priya, at\_hospital).**

**alibi(arjun, at\_home).**

**alibi(neha, at\_court).**

**alibi(ramu, at\_home).**

**relation(rajesh, priya, husband).**

**relation(rajesh, arjun, father).**

**relation(rajesh, neha, uncle).**

**relation(priya, arjun, mother).**

**relation(ramu, rajesh, servant).**

**weapon\_access(priya, yes).**

**weapon\_access(arjun, yes).**

**weapon\_access(neha, no).**

**weapon\_access(ramu, yes).**

**behavior(priya, composed).**

**behavior(arjun, nervous).**

**behavior(neha, concerned).**

**behavior(ramu, anxious).**

**strong\_suspect(Person) :-**

**suspect(Person),**

**motive(Person, \_),**

**weapon\_access(Person, yes).**

**primary\_culprit(Person) :-**

**strong\_suspect(Person),**

**alibi(Person, at\_home),**

**behavior(Person, anxious),**

**relation(Person, rajesh, servant).**

**accomplice(Person) :-**

**strong\_suspect(Person),**

**alibi(Person, at\_home),**

**behavior(Person, nervous).**

**innocent(Person) :-**

**suspect(Person),**

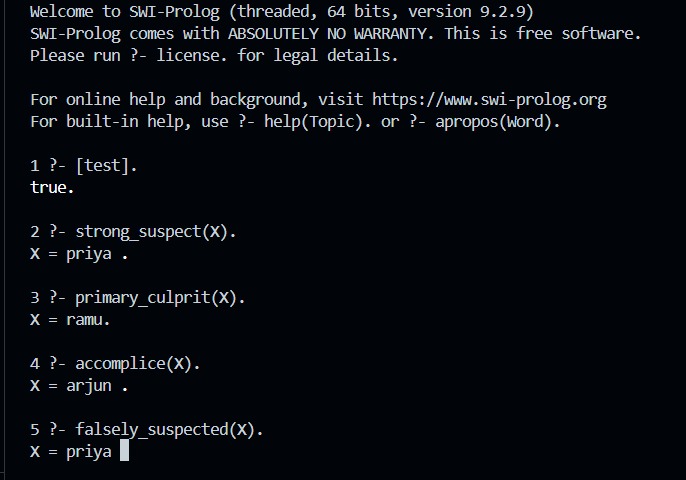
**\+ strong\_suspect(Person).**

**falsely\_suspected(Person) :-**

**suspect(Person),**

**\+ primary\_culprit(Person),**

**\+ accomplice(Person).**

****

**Post Lab Objective Questions**

**1. Which is not a Goal-based agent?**

1. Inference
2. Search
3. Planning
4. Conclusion
5. Dynamic search.

**Answer:**

**2. Which were built in such a way that humans had to supply the inputs and  
interpret the outputs?**

1. Agents
2. Sensor
3. AI System
4. Actuators

**Answer:**

**Post Lab Subjective Questions**

**Explain the role of PEAS and task environment in choosing the agent architecture. Justify your answer with an example.**

Performance measure, Environment, Actuators, and Sensors is referred to as PEAS.  
The task environment must always be adequately specified before beginning to develop an agent.  
 The task environment aids in determining whether or not the setting is  
 • Discrete / Continuous - An environment is discrete if there are just a few clearly defined, different states of it (such as in chess); otherwise, it is continuous (For example, driving).  
 • One agent / Many agents - Other agents, which may be of the same or a different sort as the agent, may be present in the environment.  
 • Accessible / Inaccessible - Whether an environment is accessible or inaccessible depends on whether the agent's sensory apparatus can access the entire state of the environment.  
 • Deterministic / Non-deterministic - Environments can be either deterministic or non-deterministic depending on whether the next state of the environment is entirely determined by the present state and the activities of the agent.  
 • Episodic / Non-episodic − In an episodic environment, each episode consists of the agent perceiving and then acting. The quality of its action depends just on the episode itself. Subsequent episodes do not depend on the actions in the previous episodes. Episodic environments are much simpler because the agent does not need to think ahead.  
 • Once the environment has been fully understood, the appropriate agent architecture can be chosen based on the characteristics of the environment.  
 • Observable / Partly Observable – A system is observable if it is feasible to infer from the precepts the whole state of the environment at each time point; otherwise, it is only partially observable.  
 • Static / Dynamic -The environment is static if it stays the same while an agent is acting; otherwise, it is dynamic.