A Laboratory Manual for

# Artificial Intelligence
# (3170716)

## B.E. Semester 7
## (Computer Engineering)


**Government Engineering College, Rajkot**

**A.Y. 2023 – 24**

# Government Engineering College, Rajkot

## Department of Computer Engineering


# Certificate

This is to certify that Mr./Ms _____

Enrolment No. _____ of B.E. Semester 7th Computer Engineering of this Institute (GTU Code: 020) has satisfactorily completed the Practical / Tutorial work for the subject **Artificial Intelligence (3170716)** for the academic year 2022-23.

Place: _____

Date: _____

**Name and Sign of Faculty member**

**Head of the Department**

# Preface

Main motto of any laboratory/practical/field work is for enhancing required skills as well as creating ability amongst students to solve real time problem by developing relevant competencies in psychomotor domain. By keeping in view, GTU has designed competency focused outcome-based curriculum for engineering degree programs where sufficient weightage is given to practical work. It shows importance of enhancement of skills amongst the students and it pays attention to utilize every second of time allotted for practical amongst students, instructors and faculty members to achieve relevant outcomes by performing the experiments rather than having merely study type experiments. It is must for effective implementation of competency focused outcome-based curriculum that every practical is keenly designed to serve as a tool to develop and enhance relevant competency required by the various industry among every student. These psychomotor skills are very difficult to develop through traditional chalk and board content delivery method in the classroom. Accordingly, this lab manual is designed to focus on the industry defined relevant outcomes, rather than old practice of conducting practical to prove concept and theory.

By using this lab manual students can go through the relevant theory and procedure in advance before the actual performance which creates an interest and students can have basic idea prior to performance. This in turn enhances pre-determined outcomes amongst students. Each experiment in this manual begins with competency, industry relevant skills, course outcomes as well as practical outcomes (objectives). The students will also achieve safety and necessary precautions to be taken while performing practical.

This manual also provides guidelines to faculty members to facilitate student centric lab activities through each experiment by arranging and managing necessary resources in order that the students follow the procedures with required safety and necessary precautions to achieve the outcomes. It also gives an idea that how students will be assessed by providing rubrics.

Artificial Intelligence (AI) has become an integral part of many industries and fields, impacting human life in numerous ways. AI techniques, such as Predicate Logic, Production Rules, and Semantic Networks, are used to encode knowledge in computer systems and solve real-world problems. Additionally, fields of AI like Game Playing, Natural Language Processing, and Connectionist Models play a vital role in various industries. It is essential for students to learn programming languages for AI as it is used in both technical and non-technical fields. AI has been implemented in every branch of engineering, making systems more effective and dynamic. The Fundamentals of Artificial Intelligence course aims to provide exposure to the basic AI techniques.

Utmost care has been taken while preparing this lab manual however always there is chances of improvement. Therefore, we welcome constructive suggestions for improvement and removal of errors if any.

## Practical – Course Outcome matrix

| Course Outcomes (COs): | |
|---|---|
| CO-1 | Understand the search technique procedures applied to real world problems |
| CO-2 | Understand and use various types of logic and knowledge representation schemes. |
| CO-3 | Understand various Game Playing techniques and apply them in programs. |
| CO-4 | Gain knowledge in AI Applications and advances in Artificial Intelligence |
| CO-5 | Use Prolog Programming language using predicate logic |

| Sr. No. | Objective(s) of Experiment | CO1 | CO2 | CO3 | CO4 | CO5 |
|---|---|---|---|---|---|---|
| 1. | Write a Prolog program which contains three predicates: male, female, parent. Make rules for following family relations: father, mother, grandfather, grandmother, brother, sister, uncle, aunt, nephew and niece. | | √ | | | √ |
| 2. | Write a Prolog program to implement Water Jug Problem. | √ | | | | √ |
| 3. | Solve 8 Puzzle Problem using A* Algorithm in any programming Language. | √ | | | | |
| 4. | Convert the given Prolog predicates into Semantic Net. cat(tom). cat(cat1). mat(mat1). sat_on(cat1,mat1). bird(bird1). caught(tom,bird1). like(X,cream) :– cat(X). mammal(X) :– cat(X). has(X,fur) :– mammal(X). animal(X) :– mammal(X). animal(X) :– bird(X). owns(john,tom). is_coloured(tom,ginger). | | √ | | | |
| 5. | Construct the Conceptual Dependency for the given statements.<br>1) John gives Mary a book<br>2) John gave Mary the book yesterday. | | √ | | | |
| 6. | Implement the Minimax algorithm for a simple tic-tac-toe game using Python Language. | | | √ | | |
| 7. | Implement Bayesian Networks algorithm for the Monty Hall Problem using any programming Language. | | | | √ | |
| 8. | Demonstrate Connectionist Model using Tool. | | | | √ | |
| 9. | Implement Genetic Algorithm using any programming Language. | | | | √ | |

| | | | | | | |
|---|---|---|---|---|---|---|
| 10. | Write a PROLOG program based on list:<br>1. To find the length of a list.<br>2. To sum all numbers of list.<br>3. To find whether given element is a member of a list.<br>4. To append the list.<br>5. To reverse the list.<br>6. To find the last element of a list.<br>7. To delete the first occurrence of an element from a list.<br>8. To delete every occurrence of an element from a list.<br>9. To find $N^{th}$ element from the list. | | | | | √ |

## Instructions for Students

1. Students are expected to carefully listen to all the theory classes delivered by the faculty members and understand the COs, content of the course, teaching and examination scheme, skill set to be developed etc.
2. Students will have to perform experiments as per practical list given.
3. Students have to show output of each program in their practical file.
4. Students are instructed to submit practical list as per given sample list shown on next page.
5. Student should develop a habit of submitting the experimentation work as per the schedule and s/he should be well prepared for the same.

## Common Safety Instructions

- Switch on the PC carefully (not to use wet hands)
- Shutdown the PC properly at the end of your Lab
- Carefully handle the peripherals (Mouse, Keyboard, Network cable etc).
- Use Laptop in lab after getting permission from Teacher

# INDEX

## (Progressive Assessment Sheet)

| Sr. No. | Objective(s) of Experiment | Page No. | Date of performance | Date of submission | Assessment Marks | Sign. of Teacher with date | Remarks |
|---|---|---|---|---|---|---|---|
| 1. | Write a Prolog program which contains three predicates: male, female, parent. Make rules for following family relations: father, mother, grandfather, grandmother, brother, sister, uncle, aunt, nephew and niece. | | | | | | |
| 2. | Write a Prolog program to implement Water | | | | | | |
| 3. | Solve 8 Puzzle Problem using A* Algorithm in any programming Language. | | | | | | |
| 4. | Convert the given Prolog predicates into Semantic Net.<br>cat(tom).<br>cat(cat1).<br>mat(mat1).<br>sat_on(cat1,mat1).<br>bird(bird1).<br>caught(tom,bird1).<br>like(X,cream) :– cat(X).<br>mammal(X) :– cat(X).<br>has(X,fur) :– mammal(X).<br>animal(X) :– mammal(X).<br>animal(X) :– bird(X).<br>owns(john,tom).<br>is_coloured(tom,ginger). | | | | | | |
| 5. | Construct the Conceptual Dependency for the given statements.<br>1) John gives Mary a book<br>2) John gave Mary the book yesterday. | | | | | | |
| 6. | Implement the Minimax algorithm for a simple tic-tac-toe game using Python | | | | | | |
| 7. | Implement Bayesian Networks algorithm for the Monty Hall Problem using any programming Language. | | | | | | |
| 8. | Demonstrate Connectionist Model using Tool. | | | | | | |
| 9. | Implement Genetic Algorithm using any programming Language. | | | | | | |
| 10. | Write a PROLOG program based on list:<br>1. To find the length of a list.<br>2. To sum all numbers of list.<br>3. To find whether given element is a member of a list. | | | | | | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | 4. To append the list.<br>5. To reverse the list.<br>6. To find the last element of a list.<br>7. To delete the first occurrence of an element from a list.<br>8. To delete every occurrence of an element from a list.<br>9. To find $N^{th}$ element from the list. | | | | | | |
| Total | | | | | | | |

# Experiment No: 1

Write a PROLOG program which contains three predicates: male, female, parent. Make rules for following family relations: father, mother, grandfather, grandmother, brother, sister, uncle, aunt, nephew and niece.

**Date: //Write Experiment Date here**

**Competency and Practical Skills**: Basic Understanding of Predicated and Prolog Syntax

**Relevant CO:** CO2, CO5

**Objectives:** (a) To learn different kinds of knowledge bases of Prolog Programming.
(b) To create a powerful and flexible system for representing, manipulating, and reasoning about information in a logical and structured way.

**Equipment/Instruments:** Personal Computer, SWI-Prolog Interpreter/GNU Prolog Interface

**Theory:**

There are only three basic constructs in Prolog: facts, rules, and queries. A collection of facts and rules is called a knowledge base (or a database) and Prolog programming is all about writing knowledge bases. That is, Prolog programs simply are knowledge bases, collections of facts and rules which describe some collection of relationships that we find interesting.

**Knowledge Base 1**

Knowledge Base 1 (KB1) is simply a collection of facts. Facts are used to state things that are unconditionally true of some situation of interest. For example, we can state that Mia, Jody, and Yolanda are women, that Jody plays air guitar, and that a party is taking place, using the following five facts:

```
woman(mia).
woman (jody).
woman (yolanda).
playsAirGuitar(jody).
party.
```

We can ask Prolog whether Mia is a woman by posing the query:

```
?- woman (mia).
Prolog will answer
yes
```

**Knowledge Base 2**

```
happy (yolanda).
listens 2Music(mia).
listens 2Music(yolanda): - happy (yolanda). playsAirGuitar(mia):- listens2Music(mia).
playsAirGuitar(yolanda):- listens 2Music(yolanda).
```

There are two facts in KB2, listens2Music(mia) and happy(yolanda). The last three items it contains

are rules.
Rules state information that is conditionally true of the situation of interest The part on the left hand side of the: - is called the head of the rule, the part on the right hand side is called the body. So in general rules say: if the body of the rule is true, then the head of the rule is true too. And now for the key point:
If a knowledge base contains a rule head - body, and Prolog knows that body follows from the information in the knowledge base, then Prolog can infer head. This fundamental deduction step is called modus ponens.

**Knowledge Base 3**

KB3, our third knowledge base, consists of five clauses:
happy(vincent).
listens2Music (butch). playsAirGuitar (vincent):-
listens 2Music(vincent),
happy(vincent). playsAirGuitar(butch):-
happy(butch).
playsAirGuitar(butch):-
listens2Music(butch). There are two facts, happy(vincent) and listens2Music(butch), and three rules.
KB3 defines the same three predicates as KB2 (namely happy, listens2Music, and playsAirGuitar) but it defines them differently. In particular, the three rules that define the playsAirGuitar predicate introduce some new ideas.

**Knowledge Base 4**

Here is KB4, our fourth knowledge base:
woman(mia).
woman(jody).
woman (yolanda).
loves (vincent,mia).
loves (marsellus,mia).
loves (pumpkin, honey_bunny).
loves (honey_bunny, pumpkin).

There are no rules, only a collection of facts. we're going to make use of variables. Here's an example:
?- woman(X).

Prolog answers this query by working its way through KB4, from top to bottom, trying to unify (or match) the expression woman(X) with the information KB4 contains. Now the first item in the knowledge base is woman(mia). So, Prolog unifies X with mia, thus making the query agree perfectly with this first item. (Incidentally, there's a lot of different terminology for this process: we can also say that Prolog instantiates X to mia, or that it binds X to mia .) Prolog then reports back to us as follows:
X = mia
That is, it not only says that there is information about at least one woman in KB4, it actually tells us who she is. It didn't just say yes, it actually gave us the variable binding (or variable instantiation) that led to success.

**Safety and necessary Precautions:**

Use appropriate names and arguments for the predicates.
Make sure that the rules are accurate and cover all possible cases.

**Procedure:**

1. Define the domain of the knowledge base.
2. Define the facts.
3. Define the rules.
4. Test the knowledge base.

**Observation/Program:**

<write prolog code and execute various goal statement>

**Conclusion:**

**Quiz:**(Sufficient space to be provided for the answers)

1. What is Prolog and what is it used for?

Answer:

2. What is a knowledge base in Prolog and how is it represented?

Answer:

3. How do you define facts and rules in Prolog?

Answer:

4. How do you query a Prolog knowledge base and what is the output?
Answer:

**Suggested Reference:**

1. http://lpn.swi-prolog.org/lpnpage.php?pageid=online

**References used by the students:** (Sufficient space to be provided)

**Rubric wise marks obtained:**

| Rubrics | Knowledge (2) | | Problem Recognition (2) | | Logic Building (2) | | Completeness and accuracy (2) | | Ethics (2) | | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | Good (2) | Avg. (1) | Good (2) | Avg. (1) | Good (2) | Avg. (1) | Good (2) | Avg. (1) | Good (2) | Avg. (1) | |
| Marks | | | | | | | | | | | |