AI Experiment 1

DATE:

Kareena Shah
60004210243
C'32

Aim : To select a problem statement related to AI
(1) Identify the problem
(2) PEAS description
(3) Problem Formulation

Theory :

Problem : Autonomous Vehicle Navigation
The problem of autonomous vehicle navigation involves creating AI systems that enable vehicles to autonomously navigate through real world environments
This entails developing algorithms that allow vehicle to perceive their surroundings using sensors like cameras, LiDar, radar & GPS, make decisions based on this information to drive safely, obey traffic rules, avoid obstacles & pedestrains & efficiently reach their destinations
The challenge lies in creating robust systems capable of handling diverse road conditions, unpredictable scenarios, & ensuring passenger safety throughout the journey

PEAS description

Performance Measure
The successful navigation of a vehicle from one point to another while obeying traffic rules, avoiding accidents & reaching destination safely.

FOR EDUCATIONAL USE

DATE:

Environment
Roads, traffic signals, pedestrains, other vehicles, weather conditions, road obstacles

Actuators
Steering, acceleration, braking mechanism, sensors, GPS

Sensors
Cameras, LiDAR, radar, motion sensors & other environmental sensors

Problem Formulation
(1) Initial State
(2) Actions
(3) Transition model
(4) Goal test
(5) Path Cost

Conclusion : Thus, we understood details & importance of PEAS analysis.

FOR EDUCATIONAL USE

AI Experiment 2

DATE:

Kreena Shah.
60004210243
C'32

Aim : To implement uninformed search algorithms
(1) BFS
(2) DFS
(3) DFID

Theory :
Uninformed search algorithms, blind search algorithms or brute force algorithms are used to find solutions in search space without any prior information or heuristics
The explore the search space systematically & do not consider the quality of the paths or the states they explore.
Three commonly used uninformed search algorithms are as follows :
(1) BFS
(2) DFS
(3) DFID

Breadth First Search

It explores a search tree by expanding all the child nodes at a given depth before moving to the next level.

FOR EDUCATIONAL USE

DATE:

Depth First Search (DFS)

It explores the search tree by selecting a child node & diving as deeply as possible before backtracking. Usually implemented using stack

Depth First Iterative Deepning (DFID)

DFID combines the benefits of both BFS & DFS. It performs a series of DFS searches with increasing depth limits starting from depth of 1 & incrementally increasing the depth limit until the goal state is found.

Conclusion : The choice of which uninformed search to use depends on the specific characteristics of the problem.
BFS guarantees optimality but may consume more memory, making it suitable for problems with limited search space
DFS is memory efficient, but may not grant optimality, making it useful for deep search space
DFID strikes a balance by combining memory efficiency & with the guarantee of finding the shortest path.

FOR EDUCATIONAL USE

DATE:

AI Experiment 3

Kreena Shah
60004210243
C'32

Aim : To implement A*, informed search algorithms.

Theory :

Informed search algorithms, heuristic search algorithms are designed to optimize the search process by using problem specific information or heuristics.
Unlike, uninformed search which explores the search space blindly, informed search algorithm aim to prioritize promising paths that are more likely to lead to a solution.
They do so by assigning a cost or value to each essential potential path based on a heuristic function which estimates the desirability of a particular state or node
The key components are :
(1) Heuristic function
(2) Cost Functions
(3) Open set & Closed set
(4) Priority queue

Heuristic Function
It is a domain specific estimate that assigns a value to each state or node in the search space.

FOR EDUCATIONAL USE

DATE:

Cost function
Represents the total cost incurred so far to reach a specific state from initial state

Open & Closed Set
Data structure used to keep track of states that need to be explored & have already been explored.

Priority Queue
It employs this to select the next state to explore

Conclusion: A* efficiently solves complex problems by intelligently navigating through search spaces.

FOR EDUCATIONAL USE

AI Experiment 4

DATE:

Kreena Shah
60004210243
C-32

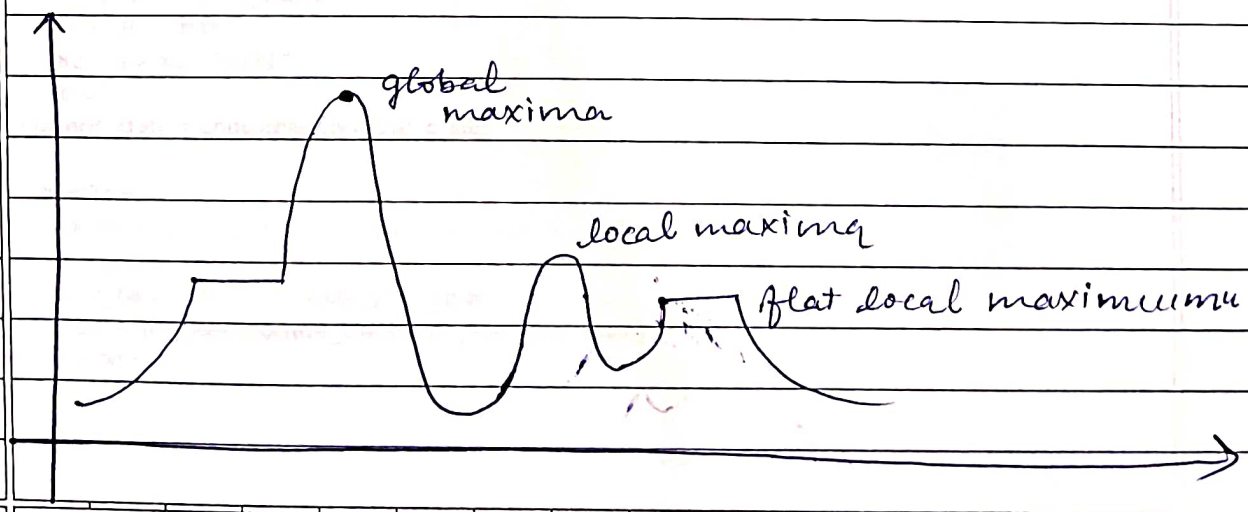
Aim : To implement local search algorithm : Hill Climbing

Theory :

Hill Climbing is a local search algorithm in AI to find the best possible solution to a problem.
It is named after the idea that it climbs the hills of search space to find the most optimal solutions but it can have limitations & may not be always find the global optimum.

Features of Hill Climbing
Generate & Test Variant
Greedy Approach
No backtracking

State Space diagram



FOR EDUCATIONAL USE

DATE:

Advantages
(1) Simplicity
(2) Efficiency

Disadvantages
(1) Limited Exploration
(2) local Optimum
(3) Dependency on initial state
(4) It does not maintain a history of visited states

Conclusion : Hill Climbing can be beneficial for monotonously increasing paths in search space but it cannot be relied on completely as it may get stuck at local minima.

DATE:

AI Experiment 5

Kreena Shah
60004210243
C'32

Aim : To implement genetic algorithm to solve optimization problems

Theory :

A genetic algorithm is a search heuristic reflecting the procen of survival of the fittest for producing the next generation.
There are primary 5 phases :
(1) Initialisation
(2) Fitness Assignment
(3) Selection
(4) Reproduction
- Crossover
- Mutation
(5) Loop through 1st 4 steps

Initialisation
A set of individuals called as population each being a solution to the given problem.

Fitness function
To determine the ability of an individual to compete with others

FOR EDUCATIONAL USE

DATE:

Selection :-
The selection of individuals for reproductions of the next generation

Reproduction
The creation of children in next generation is done
- Crossover
- Mutation

Conclusion :- Genetic algorithm can be used to find solution using local moves or renewing population with the best solutions

FOR EDUCATIONAL USE

DATE:

AI Experiment 6

Kareena Shah
60004210243
C'32

Aim : To study & implement Perceptron learning Algorithm.

Theory :

Perceptron learning is a fundamental concept in artificial machine learning & artificial intelligence particularly neural network.

Structure :
It consists of input nodes associated with weights & an output node.
The inputs & weights are multiplied & the weighted sum is passed through an activation function.
The output is a binary decision based on whether the weighted sum exceeds a certain threshold which is usually a step function.

The above process involves the following steps, mentioned below
(1) Initialization
(2) Calculation for each input
(3) Error calculation & weight input

$$\Delta w = c\,(\text{desired} - \text{output})\, i$$

$$w_{i+1} = \Delta w + w_i$$

(4) Repeat

FOR EDUCATIONAL USE

DATE:

Conclusion : Perceptron learning is a straightforward method for training a basic neural network model which while it has limitations, it played crucial role in history of neural network

FOR EDUCATIONAL USE

AI Experiment 7

DATE:

Kreena Shah
60004210243
C'32

Aim : Program to Implement family tree in Prolog

Theory :

Prolog is a logic programming language.
It has an important role in artificial intelligence.
Unlike many other programming languages, Prolog is intended primarily as a declarative programming language.
In prolog, logic is expressed as relations (called as facts & rules)
Formulation is carried out by running a query over these relations

Key features
(1) Unification
(2) Backtracking
(3) Recursion

Advantages
(1) Easy to build database
(2) Pattern matching is easy
(3) It has built-in list handling

Disadvantages
(1) LISP dominates over prolog with respect to I/O features
(2) Sometimes I/O is not easy.

FOR EDUCATIONAL USE

DATE:

Conclusion : Thus, we studied & implemented prolog.

FOR EDUCATIONAL USE

AI Experiment 8

DATE:

Koleena Shah
60004210243
C'32

Aim : Implementation on any AI problems
(1) Wumpus World
(2) Tic-tac-toe
(3) 8-queens

Theory :

8-Queens Problem

The 8-queen problem involves placing eight queens on an 8X8 chessboard in a way that no two queens threaten each other

Implementation Approach
(1) State Representation
Represent the chessboard & queens position

(2) Constraint Satisfaction
Use constraint programming / backtracking algorithms to place queens on the board while satisfying constraints

(3) Search Techniques
Employ various search strategies, such as recursive backtracking, to explore possible queen placements efficiently without violating constraints

FOR EDUCATIONAL USE

DATE:

Conclusion : Thus, we Implemented 8-queen problem

FOR EDUCATIONAL USE

AI Experiment 9

DATE:

Kreena Shah
60004210243
C'32

Aim : Identify, analyze, implement a planning problem / Rule based Expert system in a real world scenario.

Let's consider a real world scenario where a rule based expert system could be applied : Hospital Bed Management System

Problem Identification
Hospital bed management involves efficiently allocating & managing available beds for patients based on their medical needs, urgency & the hospital's resources

Analysis
A hospital has various types of beds (ICU, general, pediatric) & a fluctuating no. of patients with different medical conditions requiring admission.

Challenges
(1) Bed Allocation
(2) Optimization
(3) Dynamic Environment

Rule Based Expert System
(1) Knowledge Base
(2) Inference Engine
(3) User Interface

FOR EDUCATIONAL USE

DATE:

Rules & logic :
(1) Priority Rules
(2) Bed Availability
(3) Patient Conditions
(4) Dynamic Updates

Implementation :
(1) Data Collection :
(2) Rule Formulation
(3) System Development
(4) Testing & Validation
(5) Integration & Deployment

Outcome :
An efficient rule based expert system for hospital bed management would :
Optimize bed allocation based on patient needs & available resources
Reduce wait times for critical cases.
Provide a systematic approach for bed assignment, ensuring better patient care & resource utilization

Conclusion : Rule based expert systems contributes to improved patient outcomes & resource management in healthcare facilities

FOR EDUCATIONAL USE

Kreena Shah
60004210243

Experiment 10

Case Study - AI Application Research Paper Summary

Paper Title	Composer4Everyone: Automatic Music Generation with Audio Motif	An AI-Based Intelligent Music Composing Algorithm: Concord	Music Generation with AI Technology: Is It Possible?
Conference Name & Year	2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)	2013 International Conference on Advances in Technology and Engineering (ICATE)	2022 IEEE 5th International Conference on Electronics Technology (ICET)
Algorithm	The algorithm processes user input (humming, singing, speaking) and generates a 20-30 second piano piece. It employs expectation analysis to ensure alignment with user expectations and a rearranging system for style variations.	Concord is an object-oriented algorithm focusing on scales, chords, and note patterns. It balances musical rules and randomness to create coherent compositions, using the circle of fifths for harmony.	Reviews Biaxial-LSTM, DeepJ, and MuseGAN. These models are employed for polyphonic and pop music generation, with DeepJ focusing on style-specific generation and MuseGAN using GANs for pop music.
Dataset	The paper does not explicitly mention the dataset used for developing or training the algorithm.	No specific dataset mentioned; the algorithm is based on musical theory and object-oriented programming principles.	The paper does not detail specific datasets but reviews literature on various models and their applications in AI music generation.
Results	The application was tested with 300 participants in a listening test, showing high satisfaction with the quality and consistency of the generated music.	The paper does not provide empirical results or user testing outcomes, focusing more on the theoretical aspects of the algorithm.	The paper does not report specific results but provides an analysis of the models' capabilities and scenarios where they are applied.
Observations regarding performance	High user satisfaction, indicating effectiveness in generating music that aligns with user expectations. The system's interaction design and expectation analysis are key to its performance.	Concord's performance is not empirically tested but is theorized to produce unique and musically coherent pieces based on its algorithmic design.	The comparison of models highlights the strengths and limitations in AI music generation, emphasizing the need for further

			research in this area.
Conclusion	The app demonstrates the potential of AI in making music composition accessible to a broader audience, with promising results in user interaction and satisfaction.	Concord represents a novel approach in AI music composition, combining musical theory with computational methods, though practical applications and user experiences are yet to be explored.	The paper concludes that while AI has made significant strides in music generation, there is still a need for models that can better emulate human creativity and emotional expression.

60004210243

Page No.
Date

01/11/2023

Name : Kreena Shah
Sapid : 60004210243
Batch : Comps C'32
Subject : AI Assignment 1

(1) (i) Semantic Network

(a) A semantic network is a knowledge structure that depicts how concepts are related to one another & illustrates how they are interconnected

(b) Semantic networks use artificial intelligence (AI) programming to mine data, connect concepts & call attention to relationships.

(c) In business, this capability can make customer service more proficient by providing better product search functionality.

(d) New technologies like Microsoft Office Graph use semantic networking to bring related concepts together.

(ii)

RDF	OWL
• Resource Description Framework for describing & linking resources on web	Web Ontology language is a powerful ontology language that extends RDF
• It is primarily used for representing data in a structured & machine readable format.	It is designed to model & reason about complex relationships & ontologies in a formal & expressive way.
• RDF uses triples to express statements/facts about	OWL allows for the creation of ontologies that

SATYAM®
FOR EDUCATIONAL USE

resources	define classes, properties & relationships among data elements
• It provides a basic graph based data model for representing the information	It provides a rich set of logical constructs for defining complex concepts such as classes, individuals & properties & supports reasoning about these concepts

(2) (i) Ant Colony Optimization (ACO) is a nature inspired optimization algorithm based on the foraging behaviour of ants.

(ii) ACO is used to find solutions to complex combinatorial optimization problems

(iii) Following is the procedure of how it works

⇒ Inspiration : ACO is inspired by the way real ants find the shortest path between their nest & a food source by depositing pheromones. Ants follow paths with higher pheromone concentrations

⇒ Solution Construction : In ACO, artificial "ants" iteratively construct solutions by moving through a problem space. They make decisions based on both pheromone levels & a heuristic function that guides them toward promising solutions

⇒ Pheromone Update : After constructing a solution, ants update the pheromone levels on the paths they have traversed. Shorter & better solutions receive higher pheromone deposits.

SATYAM®
FOR EDUCATIONAL USE

Page No.
Date

→ Exploration vs Exploitation : ACO balances exploration (searching for new solutions) & exploitation (reinforcing known good solutions) by adjusting the pheromone evaporation rate & the influence of pheromone & heuristic information in decision making

→ Convergence : Overtime, ACO tends to converge towards optimal or near optimal solutions as good solutions accumulate pheromone & become more attractive to ants.

(3) Unification Algorithm for Predicate Logic

(1) The unification algorithm is used to find a substitution that makes two predicates equals in first order logic.

(2) It's crucial for reasoning & resolution in predicate logic

(3) Unification is a process of finding a common instantiation of variables in two predicates to make them the same

Eg :

Consider predicates $P(x, y)$ & $P(a, b)$. To unify them, we want to find values for x & y such that $P(x, y)$ becomes the same as $P(a, b)$.

Algorithm :

(a) Start with an empty substitution : { }

(b) Compare the arguments of the predicates

- x unifies with a, so $\{x/a\}$ to the substitution.

SATYAM®

FOR EDUCATIONAL USE

Page No.
Date

- y unifies with b, so add {x/a, y/b} to the substitution.

(c) Apply the final substitution to the original predicates to get P(a,b) & P(a,b) which are same

(4) Bayesian Belief Network {BNN}

(a) A Bayesia Belief Network is a probabilistic graphical model used for representing & reasoning about uncertainity in a system

(b) It consists of nodes representing random variables & directed edges showing probabilistic dependencies between them

(c) Bayesian Belief Network are used in various applications such as decision support & machine learning

(d) They incorporate Bayesian probability theory to update beliefs as new evidence is obtained

(5) Fuzzy Set & Operations

(a) A fuzzy set is a mathematical concept that generalizes the idea of classical sets.

(b) It allows elements to belong to a set to a degree, ranging from 0 (completely outside the set) to 1 (completely inside the set)

(c) Fuzzy sets are used in fuzzy logic to handle uncertainity

Fuzzy Set Operations :

SATYAM®
FOR EDUCATIONAL USE

Page No.
Date

(a) Union
The union of two fuzzy sets A & B at each point represents the maximum degree to which an element belongs to either A or B

(b) Intersection
The intersection of two fuzzy sets A & B at each point represents the minimum degree to which an element belongs to both A & B

(c) Complement
The complement of a fuzzy set A represents the degree to which elements do not belong to A, often denoted as "not A"

SATYAM®
FOR EDUCATIONAL USE

AI Assignment 2

DATE:

Kreena Shah
60004210243
C'32

(1) Hierarchical Planning in AI involves organizing complex planning problems into a hierarchy of tasks or actions, enabling more efficient & structured problem solving. It breaks down a large task into smaller, more manageable sub-tasks, creating a multi-level structure where higher level plans control & guide lower level plans.

Key Components :

(1) Abstraction levels
Higher level represent broader goals/tasks & lower levels details the specific actions required to achieve them.

(2) Decomposition
The process involves decomposing high level goals into sub-goals. Each sub-goal can further decompose into more detailed actions until reaching a level where actions can be directly executed.

(3) Information flow
Here, information flows in both ways in a hierarchy. Higher level plans provide guidance & constraints to lower level plans while feedback from lower level helps in refining higher level plans.

FOR EDUCATIONAL USE

DATE:

(4) Efficiency
Hierarchical Planning can be significantly reduce computational complexity by focusing on specific parts of problems at each level, leading to more efficient planning algorithms.

(2) Goal Stack Planning (GSP).
GSP is a method used in artificial intelligence & automated planning.
It involves representing a problem-solving process by breaking down complex goals into simpler subgoals to achieve a desired outcome.
It utilizes a stack based structure to organize & execute these subgoals

Example: Making Breakfast

(1) Goal: Make Pancakes
: Subgoals: Gather ingredients
Mix batter
Heat skillet
Cook pancakes
Serve pancakes

Subgoal decomposition
- Serve Pancakes
Prepare plates
Place panckakes, add topping
Serve

DATE:

- Cook Pancakes
  Pour batter on skillet
  Flip pancakes
  Ensure proper cooking time

→ Execution
The system executes each subgoal in a LIFO manner from the goal stack, working backward until the primary goal is achieved

⇒ Goal Stack Planning vs Forward State Space Planning (FSSP)

GSP	FSSP
→ Operates by decomposing a high level goal into subgoals & solving them to achieve the primary objective	Starts from initial state & works forward by applying actions to reach goal state.
→ Utilizes a stack structure to organize & manage subgoals	Represents the planning problem as a state space graph, exploring possible actions & states to reach goal
→ Operates in depth first manner, working backward from final goal through subgoal decomposition	Explores possible actions from initial state, incrementally moving towards goal state.

FOR EDUCATIONAL USE

DATE:

(3) Building an expert system has following phases

(1) Identification & Selection of Problem

The initial phase involves identifying problem domain suitable for an expert system.

This could be an area where expertise is required such as medical diagnosis, financial analysis or technical trouble shooting

(2) Knowledge Acquisition

This phase involves gathering knowledge from domain experts

(3) Knowledge Representation

Here, acquired knowledge is organized & integrated into systems.

(4) Inference Engine Development

The inference engine is the core component responsible for reasoning & making inference based on the knowledge stored.

(5) User Interface Design

It allows user to interact with expert system.

It is user friendly

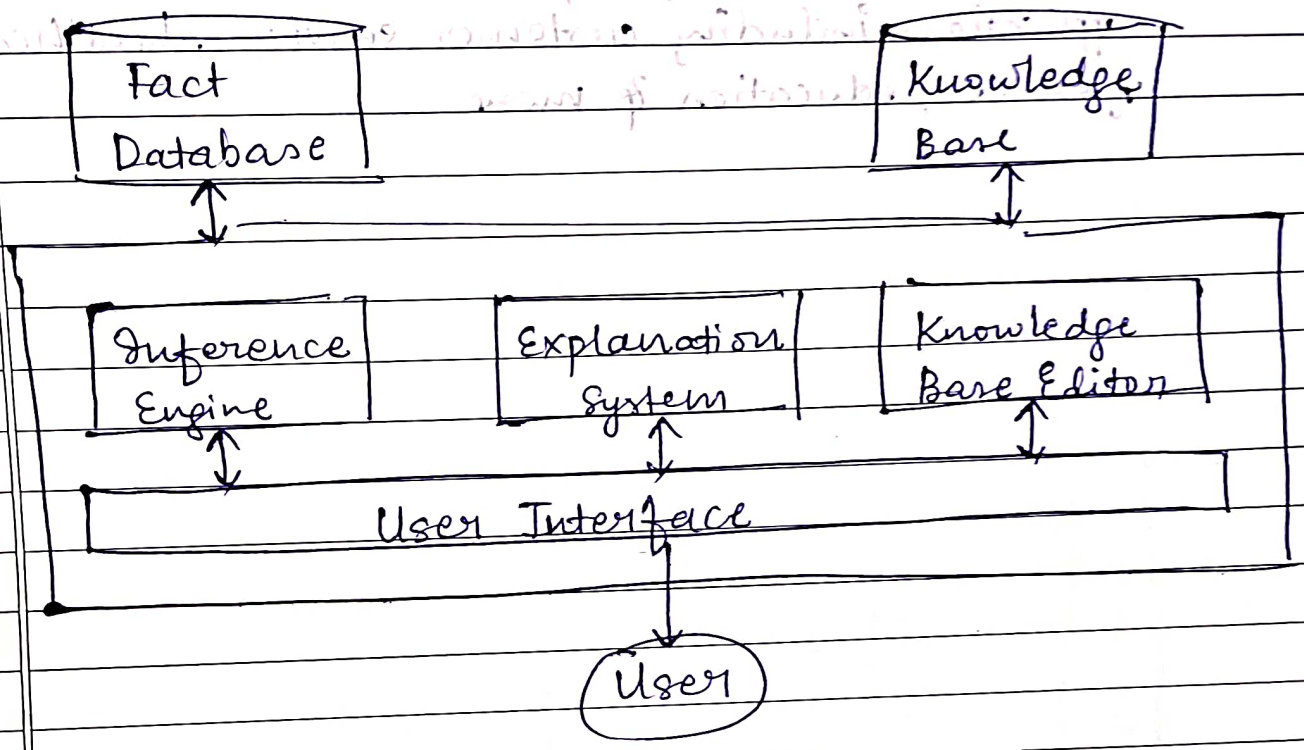
(6) Testing & Validation

Once expert system is developed, it undergoes testing & validation

FOR EDUCATIONAL USE

DATE:

(7) Deployment & Maintainance
After successful testing, expert systems are deployed for real world use



(4) We will discuss Natural Language Processing (NLP) which focuses on enabling machines to understand, interpret, generate & respond to human language in a way that is both meaningful & contextually relevant

Key components in NLP
(1) Language Understanding
(2) Machine Translation
(3) Chatbots & Conversational AI
(4) Summarization
(5) Generation

DATE:

(6) Sentiment Analysis & Opinion mining
(7) Ethical & Bias Considerations

NLP continues to play a pivotal role in various applications, including customer service, healthcare, finance, education & more